

## SOBAR for Transparent Cloud Tiering

- [Intro](#)
- [Backup Scripts](#)
- [Steps to Recover at DR Site](#)
  - [Untar the latest backup](#)
  - [Generate the guide config](#)
  - [Recreate the filesystem](#)
    - [Find the DR Descriptor](#)
    - [Build the filesystem](#)
  - [Restore policies and validate the new filesystem](#)
  - [Mount the filesystem read-only](#)
  - [Restore Metadata](#)
  - [Mount the filesystem read/write](#)
  - [Remove the mcstore directory](#)
  - [Log on to the cloud gateway](#)
  - [Change the mmsdrfs file](#)
  - [Start cloud tiering](#)
  - [Add the filesystem to tct](#)
  - [Add Cleversafe to TCT](#)
  - [Modifications to mmcloudgateway to enable premigrate](#)
  - [Copyright 2017 ArcaPix Ltd](#)

### Copyright 2017 ArcaPix Limited

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM,

OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

## Intro

This guide outlines the procedure for restoring filesystem metadata to a DR site and reestablishing the a connection with DR. The procedure is largely borrowed from "Scale-Out Backup and Restore" (SOBAR), which uses a similar methodology to restore premigrated data from a TSM/HSM enabled filesystem in the event of a total file-system loss or outage.

Prereqs:

- The DR cluster should only have TCT enabled with a TCT node predefined. No other TCT configuration should be put into place. The TCT service should not be started or running.
- The DR filesystem is normally only there to hold the image data. Part of the process requires creating a new filesystem, so the "day to day" DR filesystem should only be large enough to hold a few days worth of metadata images and related configs. Do no use all available space/LUNS for this file system.

Notices:

- Any data that is not premigrated/migrated will be lost if the primary filesystem is also lost
- This procedure should not be used to recover from a temporary outage or any other situation where the primary filesystem is recoverable.

## Backup Scripts

The following policy should be run to completion prior to every sync:

```
define(
    exclude_list,
    (
        FALSE
        OR PATH_NAME LIKE '%/.mcstore/%'
        OR PATH_NAME LIKE '%/.mcstore.bak/%'
        OR PATH_NAME LIKE '%/.ctdb/%'
        OR PATH_NAME LIKE '%/.ces/%'
        OR PATH_NAME LIKE '%/.policytmp/%'
        OR PATH_NAME LIKE '%/.images/%'
        OR PATH_NAME LIKE '%/policies/%'
    )
)

define(
    access_age,
    (DAYS(CURRENT_TIMESTAMP) - DAYS(ACCESS_TIME))
)

define(
    mb_allocated,
    (INTEGER(KB_ALLOCATED / 1024))
)
```

```

define(
    weight_expression,
    (CASE
        /*=== The file is very young, the ranking is very low ===*/
        WHEN access_age <= 1 THEN 0
        /*=== The file is very small, the ranking is low ===*/
        WHEN mb_allocated < 1 THEN access_age
        /*=== The file is resident and large and old enough,
        the ranking is standard ===*/
        ELSE mb_allocated * access_age
    END)
)

/* Define an external pool for the off-line storage */
RULE EXTERNAL POOL 'mcstore' EXEC '/usr/lpp/mmfs/bin/mmcloudgateway
files' OPTS '-F'

/* Define migration rule with a threshold to trigger lowspace events
and move data to the external off-line pool. When on-line usage
exceeds 25% full it will move the coldest files to off-line storage
until the on-line usage is reduced to 20% utilization level. Only files
that have
    data on-line are eligible for migration. */
RULE 'MoveOffline' MIGRATE FROM POOL 'satal'
                                THRESHOLD(0,75,0)
                                WEIGHT(weight_expression)
                                TO POOL 'mcstore'
                                WHERE NOT(exclude_list)
/* Define default placement rule */

/* MACRO NAMES MUST BE DIFFERENT TO FILESET NAMES */

define(nas1, satal)

/* TCT RECALL RULES */
RULE 'OpenRead'
EVENT 'OPEN_READ'
ACTION(System('/opt/ibm/MCStore/bin/mcstore recall -c -i ' ||
varchar(INODE) || ' -g ' || varchar(GENERATION) || ' -s 0' || ' -f ' ||
varchar(FS_ID)) = 0)
WHERE(XATTR('dmapi.MCEA', 5, 1) == 'N')
RULE 'else' EVENT 'OPEN_READ' DIRECTORIES_PLUS

RULE 'OpenWrite'
EVENT 'OPEN_WRITE'
ACTION(System('/opt/ibm/MCStore/bin/mcstore recall -c -i ' ||
varchar(INODE) || ' -g ' || varchar(GENERATION) || ' -s 0' || ' -f ' ||
varchar(FS_ID)) = 0)
WHERE(XATTR('dmapi.MCEA', 5, 1) == 'N')
RULE 'else' EVENT 'OPEN_WRITE' DIRECTORIES_PLUS

/* PLACEMENT RULES */

RULE 'nas-placement' SET POOL 'nas1' WHERE FILESET_NAME LIKE 'nas-%'

RULE 'default' SET POOL 'nas1'

```

**Code Block 1 Execute the Following Policy (from running policy)**

PREMIGRATE, is not supported by 'mmcloudgateway files' until SS 4.2.3

Below is the appypolicy method for the above policy

```
## The actual policy execution for premigration (should run in main DR
script after /mmfs1 'zeroed'

export USED_INODES=`df -i /mmfs1 | grep -v Mounted | awk '{print $3}'`
if [ $USED_INODES -lt 1000000 ]
    then export BUCKET_SIZE = 2
else

    export BUCKET_SIZE=`echo "$USED_INODES / 1000000" | bc`
fi

/usr/lpp/mmfs/bin/mmapplypolicy mmfs1 -N arcapix-001-hsm-001,arcapix-001-
sn-001,arcapix-001-sn-002 -a 16 -A $BUCKET_SIZE -m 3 -g /mmfs1/.policytmp
-s /mmfs1/.policytmp
```

## Code Block 2 The Policy Execution

Below is a script for automating the majority of the process. This should be run from arcapix-001-hsm-001

```
#Copyright 2017 ArcaPix Ltd

#Permission is hereby granted, free of charge, to any person obtaining a
copy of this software and associated documentation files (the
"Software"), to deal in the Software #without restriction, including
without limitation the rights to use, copy, modify, merge, publish,
distribute, sublicense, and/or sell copies of the Software, and to permit
#persons to whom the Software is furnished to do so, subject to the
following conditions:

#The above copyright notice and this permission notice shall be included
in all copies or substantial portions of the Software.

#THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
MERCHANTABILITY, FITNESS FOR A PARTICULAR #PURPOSE AND NONINFRINGEMENT.
IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT
OR #OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR
THE USE OR OTHER DEALINGS IN THE SOFTWARE.

#!/bin/bash

# Grab date the will be used for the process
CURRENT_DATE=`date +"%m-%d-%y-%H-%M-%S"`

# Get the current number of inodes in use. We will use this to tune the
number of buckets during the policy sort and image generation
USED_INODES=`df -i /mmfs1 | grep -v Mounted | awk '{print $3}'`

# Determine the number of buckets that will be used for the policy sort
and image generation. If less than 1m, set to '2'
if [ $USED_INODES -lt 1000000 ]
    then BUCKET_SIZE=2
else

    BUCKET_SIZE=`echo "$USED_INODES / 1000000" | bc`
```

```

fi

# Take a snapshot which we will use to generate and copy data from
/usr/lpp/mmfs/bin/mmcrsnapshot mmfs1 img

# Create the directory that we will use to store the backup in
mkdir /mmfs1/.images/${CURRENT_DATE}

# Create a backup of the filesystem and cluster configuration and store
it in the backup directory
/usr/lpp/mmfs/bin/mmbackupconfig mmfs1 -o
/mmfs1/.images/${CURRENT_DATE}/mmfs1.backup.config
/usr/lpp/mmfs/bin/mmclsconfig >
/mmfs1/.images/${CURRENT_DATE}/cluster.backup.config
cp -rp /var/mmfs /mmfs1/.images/${CURRENT_DATE}/var_mmfs

# Take a copy of the FSROOT/.mcstore directory, which contains the cloud
databases
cp -rp /mmfs1/.snapshots/img/.mcstore /mmfs1/.images/${CURRENT_DATE}

# Take a copy of the current TCT configuration
mkdir /mmfs1/.images/${CURRENT_DATE}/var_MCStore
cp -rp /var/MCStore/.mcstore_settings
/mmfs1/.images/${CURRENT_DATE}/var_MCStore
cp -rp /var/MCStore/.mcstore.jceks
/mmfs1/.images/${CURRENT_DATE}/var_MCStore

# Run the image backup. This will use /mmfs1/.policytmp as the 'staging'
area. We will also use the img snapshot
# For the policy sort and image generation phase of the executed policy,
we will use 20 sort threads with
# $BUCKET_SIZE for the number of buckets. TSM will not be used to back up
the files
/usr/lpp/mmfs/bin/mmimgbackup mmfs1 -g /mmfs1/.policytmp --image
mmfs1_${CURRENT_DATE} --notsm -S img -N arcapix-001-sn-001,arcapix-001-
sn-002,arcapix-001-hsm-001,arcapix-001-mn-001,arcapix-001-mn-002 -a 20 -A
$BUCKET_SIZE -s /mmfs1/.policytmp

# copy the images to .images directory where we are storing the rest of
the backup
mv /mmfs1/.policytmp/imgbackup_* /mmfs1/.images/${CURRENT_DATE}

# tar up the current image directory
/usr/bin/tar cvf /mmfs1/.images/backup_${CURRENT_DATE}.tar
/mmfs1/.images/${CURRENT_DATE}

# Checksum the tar ball for future checks
/usr/bin/md5sum /mmfs1/.images/backup_${CURRENT_DATE}.tar >
/mmfs1/.images/checksum_${CURRENT_DATE}

# sync the backup and checksum to DR
/usr/bin/rsync -av /mmfs1/.images/backup_${CURRENT_DATE}.tar
rsync://172.29.4.20/drstaging
/usr/bin/rsync -av /mmfs1/.images/checksum_${CURRENT_DATE}
rsync://172.29.4.20/drstaging

# remove the backup from primary
rm -rf /mmfs1/.images/*

# Delete the snapshot created to generate the backup
/usr/lpp/mmfs/bin/mmdelsnapshot mmfs1 img

```

**Code Block 3 Sync to DR Script**

## Steps to Recover at DR Site

This procedure involved building the new filesystem (which should not preexist), restoring the metadata and re-establishing the link with Cleversafe.

### Untar the latest backup

The backups can be found in the root of /drstaging.

```
cd /drstaging
root@arcapix-001-dr-sn-001:/drstaging# ls -lrt
total 134132736
-rw-r--r-- 1 root root 137351833600 Feb  6 23:55 backup_02-06-17-23-01-43.tar
-rw-r--r-- 1 root root              78 Feb  6 23:59 checksum_02-06-17-23-01-43

root@dr-arcapix-001-dr-sn-001:/drstaging# tar xvf backup_02-06-17-23-01-43.tar
```

### Generate the guide config

The following will create a file, using the 'mmfs1.backup.config' file generated on primary, which contains a human readable representation of the filesystem config at DR. The '-F mmfs1\_restore.out' argument is the target file.

You can use this file to grab the correct 'mmcrfs' syntax to rebuild the DR filesystem to match the settings as closely as possible to production.

```
root@arcapix-001-dr-sn-001:/drstaging/mmfs1/.images/02-06-17-23-01-43#
mmrestoreconfig mmfs1 -i mmfs1.backup.config -F mmfs1_restore.out
mmrestoreconfig: Configuration file successfully created in
mmfs1_restore.out
mmrestoreconfig: Command successfully completed
```

#### Code Block 4 Create Filesystem Config Restore File

## Recreate the filesystem

### Find the DR Descriptor

On 'arcapix-001-dr-sn-001', browse to the ArcaPix config directory. This directory will contain the required descriptor file to rebuild the filesystem and is located in /opt/arcapix/usr/share/configs. The correct descriptor file is "mmfs1.descFile".

In case this file is lost, it's contents should be:

```
%nsd:
    nsd=e5600_001_L000
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=metadataOnly
    failureGroup=1
    pool=system

%nsd:
    nsd=e5600_001_L001
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=metadataOnly
    failureGroup=1
    pool=system

%nsd:
    nsd=e5600_001_L002
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=metadataOnly
    failureGroup=1
    pool=system

%nsd:
    nsd=e5600_001_L003
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=metadataOnly
    failureGroup=1
    pool=system

%nsd:
    nsd=e5600_001_L005
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_001_L006
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_001_L007
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_001_L008
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=2
    pool=sata1
```

```

%nsd:
    nsd=e5600_001_L009
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_001_L010
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_001_L011
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=2
    pool=sata1

%nsd:
    nsd=e5600_002_L000
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=metadataOnly
    failureGroup=3
    pool=system

%nsd:
    nsd=e5600_002_L001
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=metadataOnly
    failureGroup=3
    pool=system

%nsd:
    nsd=e5600_002_L002
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=metadataOnly
    failureGroup=3
    pool=system

%nsd:
    nsd=e5600_002_L003
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=metadataOnly
    failureGroup=3
    pool=system

%nsd:
    nsd=e5600_002_L004
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L005
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L006

```



```

        servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
        usage=dataOnly
        failureGroup=4
        pool=sata1

%nsd:
    nsd=e5600_002_L007
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L008
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L009
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L010
    servers=arcapix-001-dr-sn-001,arcapix-001-dr-sn-002
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd:
    nsd=e5600_002_L011
    servers=arcapix-001-dr-sn-002,arcapix-001-dr-sn-001
    usage=dataOnly
    failureGroup=4
    pool=sata1

%nsd: device=/dev/sda2
    nsd=descOnly1
    servers=arcapix-001-dr-sn-001
    usage=descOnly
    failureGroup=101
    pool=system

%nsd: device=/dev/sda2
    nsd=descOnly2
    servers=arcapix-001-dr-sn-002
    usage=descOnly
    failureGroup=102
    pool=system

```

## Build the filesystem

Using the 'mmcrfs' line in the generated output from the 'mmrestoreconfig' step above, recreate the filesystem. You may need to append '-i 512' if this was not contained within the output.

```

root@arcapix-001-dr-sn-001:/opt/arcapix/usr/share/configs# mmcrfs mmfs1 -
F mmfs1.descFile -m 2 -R 3 -j scatter -k nfs4 --metadata-block-size 65536
-B 4194304 --filesetdf --version 4.2.2.0 -L 4194304 --perfileset-quota -
T /mmfs1 -t Z --inode-limit 134217728:500736 -Q yes -i 512
Warning: file system is not 4K aligned due to small metadata block size:
65536; metadata subblock size: 2048.
Native 4K sector disks cannot be added to this file system unless the
disk that is used is dataOnly and the data block size is at least 128K.
One or more 512 byte emulation disk(s) are being added to a non 4k
aligned file system. This may impact performance.

The following disks of mmfs1 will be formatted on node arcapix-001-dr-sn-
002:
    e5600_001_L000: size 757464 MB
    e5600_001_L001: size 757464 MB
    e5600_001_L002: size 757464 MB
    e5600_001_L003: size 757464 MB
    e5600_001_L005: size 60293120 MB
    e5600_001_L006: size 60293120 MB
    e5600_001_L007: size 60293120 MB
    e5600_001_L008: size 60293120 MB
    e5600_001_L009: size 60293120 MB
    e5600_001_L010: size 60293120 MB
    e5600_001_L011: size 60207104 MB
    e5600_002_L000: size 757464 MB
    e5600_002_L001: size 757464 MB
    e5600_002_L002: size 757464 MB
    e5600_002_L003: size 757464 MB
    e5600_002_L004: size 60293120 MB
    e5600_002_L005: size 60293120 MB
    e5600_002_L006: size 60293120 MB
    e5600_002_L007: size 60293120 MB
    e5600_002_L008: size 60293120 MB
    e5600_002_L009: size 60293120 MB
    e5600_002_L010: size 60293120 MB
    e5600_002_L011: size 60207104 MB
Formatting file system ...
Disks up to size 6.6 TB can be added to storage pool system.
Disks up to size 935 TB can be added to storage pool sata1.
Creating Inode File
Creating Allocation Maps
Creating Log Files
Clearing Inode Allocation Map
Clearing Block Allocation Map
Formatting Allocation Map for storage pool system
Formatting Allocation Map for storage pool sata1
    85 % complete on Mon Feb  6 23:28:10 2017
    100 % complete on Mon Feb  6 23:28:11 2017
Completed creation of file system /dev/mmfs1.
mmcrfs: Propagating the cluster configuration data to all
affected nodes. This is an asynchronous process.

```

**Do not mount the filesystem read/write. Doing so will set a flag on the filesystem which will prevent you from restoring and the filesystem will have to be rebuilt.**

## Restore policies and validate the new filesystem

Browse to the untarred backup and run the following. If there are any issues with the newly created filesystem, they will be flagged up in this step. This will also insert the running policy from production into the new filesystem.

```
cd /drstaging/mmfs1/.images/02-06-17-23-01-43

root@arcapix-001-dr-sn-001:/drstaging/mmfs1/.images/02-06-17-23-01-43#
mmrestoreconfig mmfs1 -i mmfs1.backup.config --image-restore
mmrestoreconfig: Quota is enabled.
mmrestoreconfig: Disabling quota ...
mmchfs: Propagating the cluster configuration data to all
         affected nodes. This is an asynchronous process.
-----
Configuration restore of mmfs1 begins at Tue Feb  7 09:27:41 GMT 2017.
-----
mmrestoreconfig: Checking disk settings for mmfs1:
mmrestoreconfig: Checking the number of storage pools defined for mmfs1.
mmrestoreconfig: Checking storage pool names defined for mmfs1.
mmrestoreconfig: Checking storage pool size for 'system'.
mmrestoreconfig: Checking storage pool size for 'sata1'.
mmrestoreconfig: Checking filesystem attribute configuration for mmfs1:
mmrestoreconfig: Checking fileset configurations for mmfs1:

mmrestoreconfig: Checking policy rule configuration for mmfs1:
Restoring backed up policy file.
Validated policy 'policyfile.backup': Parsed 8 policy rules.
Policy 'policyfile.backup' installed and broadcast to all nodes.
mmrestoreconfig: Command successfully completed
```

## Mount the filesystem read-only

```
mmmount mmfs1 -o ro -a
```

## Restore Metadata

Run the following to restore the metadata to the filesystem. This step is likely to take several hours and should be run in a 'screen' session. Replace "02-06-17-23-01-43/imgbackup\_161660/mmPolicy.161843.5AD4A2D0" with the location of the metadata restore files (\*.sbr).

```
root@arcapix-001-dr-sn-002:/drstaging/mmfs1/.images/02-06-17-23-01-43#
mmimgrestore mmfs1 /drstaging/mmfs1/.images/02-06-17-23-01-
43/imgbackup_161660/mmPolicy.161843.5AD4A2D0 -N arcapix-001-dr-sn-
001,arcapix-001-dr-sn-002 -s /drstaging/.policytmp -g
/drstaging/.policytmp
```

## Mount the filesystem read/write

```
mmumount mmfs1 -a
mmmumount mmfs1 -a
```

## Remove the mcstore directory

Remove out the restored mcstore directory located in /mmfs1/.mcstore

```
rm -rf /mmfs1/.mcstore
```

## Log on to the cloud gateway

The remaining actions need to be performed from the cloud gateway server, which is 'arcapix-001-dr-nirv-001-10g'

```
ssh arcapix-001-dr-nirv-001-10g
```

## Change the mmsdrfs file

The restored filesystem already points at the tct service, which will prevent you from properly reinstating the service. Open the file /var/mmfs/gen/mmsdrfs and look for the following line:

```
mmsdrfs:%%home%%:30_SG_HEADR:mmfs1::151::::0::scatter:no:tctservers:::::
:~::~:
```

Remove the tctservers portion of the string so that it looks like:

```
mmsdrfs:%%home%%:30_SG_HEADR:mmfs1::151::::0::scatter:no:::::::::::::
```

Remove only the 'tctservers' entry. Do not edit any other part of the line or the filesystem will likely no longer function correctly

## Start cloud tiering

Start the cloud tiering service with:

```
mmcloudgateway service start
```

## Add the filesystem to tct

This will recreate the required directories under /mmfs1/.mcstore and prepare the filesystem for cloud tiering

```
mmcloudgateway filesystem create --cloud-nodeclass tctServers --  
filesystem mmfs1 --container-prefix tctvault
```

## Add Cleversafe to TCT

This is the final step. Ensure that /root/passwordfile contains the required passphrase for authenticating with the cleversafe (this file can be removed after this step)

```
mmcloudgateway account create --cloud-type CLEVERSAFE-NEW --username XXXX  
--pwd-file /root/passwordfile --cloud-url http://10.7.30.16 --cloud-name  
TestRelocate --cloud-nodeclass tctServers --enable TRUE --location  
tctdata --meta-location tctmetadata
```

**Copyright 2017 ArcaPix Ltd**