

Tivoli Storage Manager Sync Cookbook

Tivoli Storage Manager (or TSM) is IBM's enterprise backup solution, and provides both a command line interface program that can be used to query the information required for synchronization, and ODBC driver that allows you to directly query data from the TSM database.

TSM is designed around several database tables that must be queried to get a complete list of tapes. These tables are: The Volume Table. This table lists all tapes that contain active data for a given TSM system. Any Volume that become scratch and is no longer managed by the DRM subsystem will be dropped from this table. The DRM Table. This table lists all tapes are considered critical for recovery purposes. The Library Table. This table is only available on servers that are considered library masters and lists all Volumes that are in an ATL environment. This table will also include scratch tapes that are currently in an ATL.

Data Extraction

Extracting the Volume Table

Call the TSM command line `dsmadm`,
Suppress headings and pass `userid` and `password`.
Pass TSM server name.
Output in table mode to file `TSM-Volumes.csv`.
Output in CSV mode and pass query data

```
dsmadm -dataonly=yes -id=userid -pa=password -tcpserveraddress=tsm01.gazillabyte.local -  
displaymode=table -outfile=TSM-Volumes.csv -commadelimited "select  
volume_name,location,stgpool_name from volumes"
```

Extracting the DRM Table

Call the TSM command line `dsmadm`,
suppress headings and pass `userid` and `password`.
Pass TSM server name.
Output the data in table mode to file `TSM-Volumes.csv`.
Output in CSV mode and pass query data.

```
dsmadm -dataonly=yes -id=userid -pa=password -  
tcpserveraddress=tsm01.gazillabyte.local -displaymode=table -outfile=TSM-  
DRM.csv -commadelimited "select volume_name,state,stgpool_name from drmedia"
```

Synchronization



You will need to install the [TapeTrack Sync software](#) to complete these instructions.

Synchronization with TapeTrack is performed by calling the [TMSS10Sync](#) command line program, along with:

1. The CSV file produced from the DRM table
2. The CSV file produced from the Volume table
3. A synchronization definition file that instructs the program how to interpret the TSM table outputs.

Example Command Line Arguments

```
:: Stop processing if we hit a command error
#!/bin/bash

:: Set some environmental variables.
set -e
set OUTDIR=/var/tapetrack/reports
set ETC=/etc/tapetrack
set UID=TSMusername
set PW=TSMpassword
set TSMIP=tsm01.gazillabyte.local

echo "Step 1: Extracting DRM information from TSM Server: " $TSMIP

:: Connect to TSM and get the DRM table.
dsmadm -dataonly=yes -id=$UID -pa=$PW -tcpserveraddress=$TSMID -
displaymode=table -outfile=$OUTDIR/TSM-DRM.csv -commadelimited "select
volume_name,state,stgpool_name,voltype from drmedia"

echo "Step 2: Synchronizing with TapeTrack"

:: Synchronize off the DRM table.
TMSS10Sync -a -d $ETC/TSM-DRM.ttidef -S user:-
$PW@tapetrack.gazillabyte.local < $OUTDIR/TSM-DRM.csv > $OUTDIR/TSM-DRM-
Sync.stdout 2> $OUTDIR/TSM-DRM-Sync.stderr \

echo "Step 3: Extracting Volume information from TSM Server: " $TSMIP

:: Connect to TSM and get the Volume table.
dsmadm -dataonly=yes -id=$UID -pa=$PW -tcpserveraddress=$TSMID -
displaymode=table -outfile=$OUTDIR/TSM-VOL.csv -commadelimited "select
volume_name,location,stgpool_name from volumes"

echo "Step 4: Synchronizing with TapeTrack"
```

```
:: Synchronize off the Volume table.
TMSS10Sync -a -d $ETC/TSM-DRM.ttidef -S user:-
$PW@tapetrack.gazillabyte.local < $OUTDIR/TSM-VOL.csv > $OUTDIR/TSM-VOL-
Sync.stdout 2> $OUTDIR/TSM-VOL-Sync.stderr

:: Check to see if today is Monday. Move any Volume that is still scratch.
if [[ $(date +%u) -eq 1 ]] ; then
  echo TMSS10MoveScratch -S user:-$PW@tapetrack.gazillabyte.local -R "LIBR"
```

Example Synchronization Definition

```
# Set input file
SetFile("TSM-Volume.csv");
#
# Set the Customer and Media as literal values as they never change
#
SetLiteral(CUSTOMER, "ACME");
SetLiteral(MEDIA, "LTO");
#
# Set the delimiter to a CSV
#
SetCSVDelimiter(",")
#
# Set the Volume ID
Extract(VOLUME, 1, 8, 0);
#
# Set the volume description
Extract(DESCRIPTION, 3, 100, 0);
#
# get the repository value
Extract(REPOSITORY, 2, 20, 0);
# Translate repository value based off initial value
# If Repository = Vault, Courier or Courier Retrieve set Repository to OFFS
# If Repository = Anything else (Mountable, Vault Retrieve) set Repository
to LIBR
AddTranslation(REPOSITORY, "Vault", "OFFS");
AddTranslation(REPOSITORY, "Courier*", "OFFS");
AddTranslation(REPOSITORY, "*", "LIBR");
# Set all volumes to scratch
SetLiteral(SCRATCH, "false");
```

From:

<https://rtfm.tapetrack.com/> - **TapeTrack Documentation**

Permanent link:

<https://rtfm.tapetrack.com/cookbook/tivolistoragemanagement?rev=1538616388>



Last update: **2025/01/21 22:07**