# Networker Sync Cookbook

Networker

## Extracting Networker Information

First step in syncing volume information from Networker is to extract a text or csv file with the volume, barcode, family, type, location and pool by running the mminfo command.

<sxh plain;> mminfo -xc, -a -r “volume,barcode,family,type,location,pool” > report.csv </sxh>

Sample output

<sxh plain;> volume,barcode,family,type,location,pool
000000L6,000000L6,tape,LTO,Offsite,weekly_full 000001L6,000001L6,tape,LTO,Offsite,weekly_full
000002L6,000002L6,tape,LTO,Offsite,weekly_full 000003L6,000003L6,tape,LTO,Offsite,weekly_full
000004L6,000004L6,tape,LTO,Offsite,weekly_full 000005L6,000005L6,tape,LTO,Offsite,weekly_full
000006L6,000006L6,tape,LTO,Offsite,weekly_full 000007L6,000007L6,tape,LTO,Offsite,weekly_full
</sxh>

To interpret the output in the file we use a definition file (default.ttidef) which reads the data in the Networker file, line by line, and parses the information which TMSS10Sync then uses to update the volumes location (and other data as specified) in the TapeTrack Server. It is best practice to write the definition file in a program like Notepad or Vi to ensure it is plain text and has no hidden control characters.

## default.ttidef

default.ttidef file sample with code to:

- Set CSV file delimiter (2)
- Exclude any string that starts with barcode (5)
- Set customer ID to US01 (8)
- Set media ID to LTO (9)
- Set default repository to LIBR (12)
- extract volume ID from column 1, length 8 characters (15)
- extract volume description from column 6, length 50 characters (18)
- If description contains the string “Offsite” set repository to **OFFS** (21)

<sxh plain;> # Set CSV delimiter SetCSVDelimiter(“,”);

# Skip any line that begins with barcode (header) AddString(EXCLUSION, 0, “barcode*”);

# Set customer ID and media ID to sync with setLiteral(CUSTOMER, “US01”); setLiteral(MEDIA, “LTO”);

# All volumes initially set to library setLiteral(REPOSITORY, “LIBR”);

# Extract Volume ID from column 1 (up to eight characters long) Extract(VOLUME, 1, 8, 0);

# Extract volume Description from column 6 (up to fifty characters long) Extract(DESCRIPTION, 6, 50, 0);

# Send any volume from the pool **offsite** to OFFS repository AddTranslation2(REPOSITORY, DESCRIPTION, “Offsite*”, “OFFS”); </sxh>

## Sync Information

Create a batch file with code to run Networkers mminfo command and then TMSS10Sync with the required parameters to input the report.csv file and default.ttidef file. Running with the -M command will place the Sync in dump mode, which allows you to check the output from the process without executing any alterations in the TapeTrack Server.

Sample batch file to run mminfo, pass the information to TMSS10Sync and output a dump of information. <sxh plain;> mminfo -xc, -a -r “volume,barcode,family,type,location,pool” > report.csv

TMSS10Sync -M < report.csv > stdout.txt 2> stderr.txt </sxh>

When you are happy with the output, remove the -M attribute and insert -S attribute and access credentials for the TMSS10Sync command line. Adding an attribute -a will add any new volumes in Networker not present in TapeTrack.

<sxh plain;> mminfo -xc, -a -r “volume,barcode,family,type,location,pool” > report.csv

TMSS10Sync -S user:-password@server -a < report.csv > stdout.txt 2> stderr.txt </sxh>

## Automating The Sync Process

Once the Sync process is set up and running correctly, the batch file can be run automatically every hour (or at a time frame that suits) through windows scheduler to keep TapeTrack up to date.

From:
https://docs.tapetrack.com/ - **TapeTrack Documentation**

Permanent link:
**https://docs.tapetrack.com/cookbook/networker?rev=1537324502**

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