Info

First of all, this program only works for Windows with Framework 4.0 installed.
Since the backup drives or original drives in Olive are based on Linux, we need a tool to correcty read the drives. Via http://www.ext2fsd.com/ a program can be downloaded that can acces the drive via windows and assign a drive letter to it.

The songs can now be obtained from the AudioLibrary folder and the Album Images from the Thumbnails folder.

In the root, there is also a file called "zDB.zc". This is the actual database, only encrypted.
I found out how the encryption worked, however, someone else was first so I give him the credits: https://github.com/dogshoes/zcryp

The database is in SQLite format, therefore the tools to access it where downloaded by https://system.data.sqlite.org

If you want to look at the database and it's contents I can recommend
http://sqlitebrowser.org/
or an add-on for Firefox: https://addons.mozilla.org/nl/thunderbird/addon/sqlite-manager/

Export the Backup drive

Now the program!. First create a folder and copy the files in the zip file to that folder and start the program "Olive Backup Exporter.exe":

Fill in the location of the backup drive (default is "G:\", but it should be the driveletter that you've assigned it to with Ext2Fsd). The Target Location is automatically filled in during startup but can also be changed. Sorry, no fancy tools to select the correct path, just a textbox…
Make sure that the target location has enough free space to hold all the music and folders!

If you use a network storage, mount the drive to a specific drive letter. Network path's are not recognized! It is recommended to leave the "To Location" path as per default.

First decrypt the database with the Decrypt and Import Database button. Depending on the size this can take a few seconds, so please be patient. A file "zdb" is now created, it is the decrypted Olive database. The contentsof the database can now be seen by any SQLite browser.

Now press the GO! Button. All the audio files are copied to the provided location as ARTIST\ALBUM\TRACK – TITLE. The Album images are saved as ARTIST\ALBUM\FOLDER

Please be aware that tags are not saved! But with a program like MP3Tag the tags can be filled automatically. See http://www.mp3tag.de/en/

Export the database Meta tagging

Export of the meta data is technically not difficult. However, the main problem was to choose which data and to what kind of files (WAV files do not contain tags). Therefore I decided to make a beta version of my program Log Converter (see www.lydo.nl) with SQLite incorporated.

This program uses pseudocode in a textfile to change data. With the aid of the Log Converter manual and the examples, everyone can export the required meta tags to a textfile. The textfile can then be used for import via dedicated programs as mp3Tag or even as a batch file to run with other external programs.

Start the program "Log Converter 2.1B.exe".
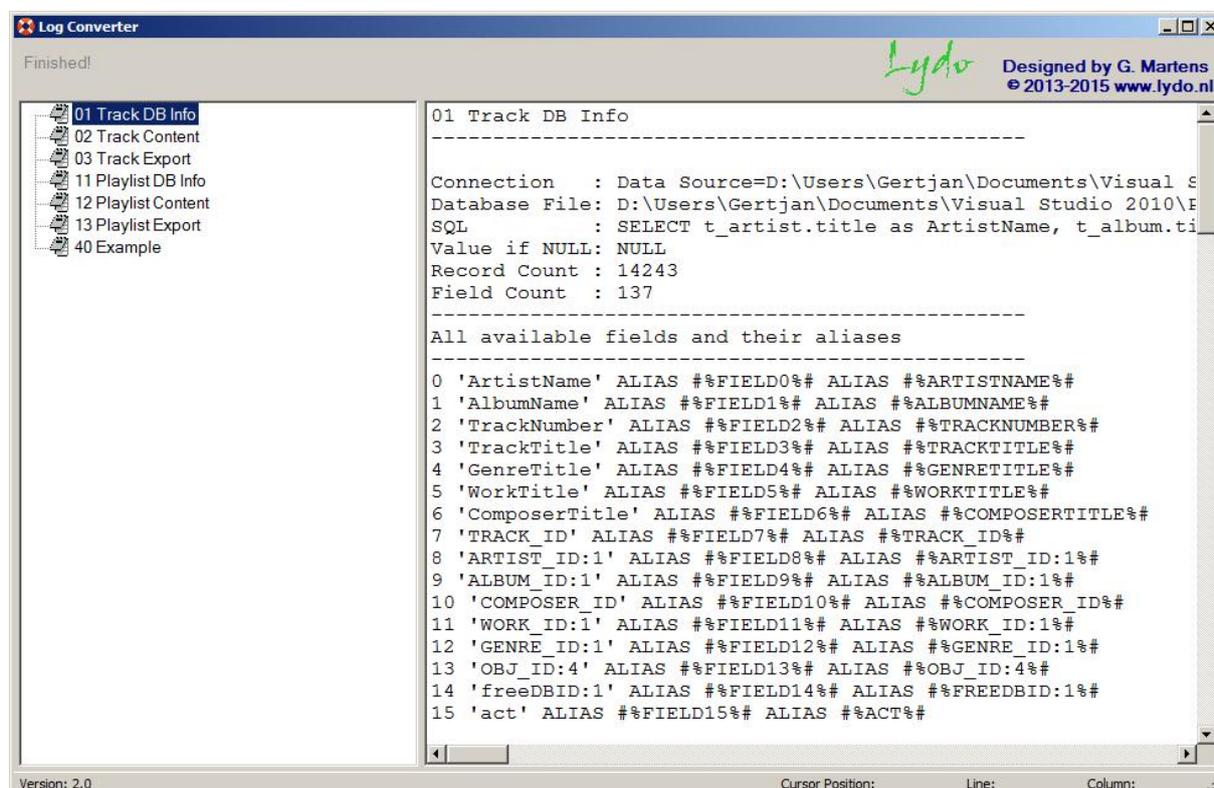
To get the data from a database, a connection has to be made to the file and then a query has to be made via a language called SQL. In "40 Example" (it is just a textfile with pseudo code) a query "SELECT * FROM t_genre LIMIT 0, 5" is made. In plain english, it asks for all fields from the table t_genre but the output is limited to the first 5 records. If you remove the last LIMIT part, all records in that table will be shown. Not really difficult, but Olive did make a mess of it's database with duplicate information stored in different tables…

Therefore I found the following query to obtain all the track information:

```
SELECT t_artist.title as ArtistName, t_album.title as
AlbumName, t_track.trackNo as TrackNumber, t_object.title as
TrackTitle, t_genre.title as GenreTitle, t_work.title as
WorkTitle, t_composer.title as ComposerTitle, t_track.*,
t_artist.*, t_album.*, t_object.*, t_genre.*, t_work.*,
t_composer.* FROM (t_composer INNER JOIN (t_work INNER JOIN
(t_genre INNER JOIN (t_album INNER JOIN (t_artist INNER JOIN
t_track ON t_artist.OBJ_ID = t_track.ARTIST_ID) ON
t_album.OBJ_ID = t_track.ALBUM_ID) ON t_genre.OBJ_ID =
t_track.GENRE_ID) ON t_track.WORK_ID = t_work.OBJ_ID) ON
t_track.COMPOSER_ID = t_composer.OBJ_ID) INNER JOIN t_object
ON t_track.OBJ_ID=t_object.OBJ_ID ORDER BY ArtistName,
AlbumName,  TrackNumber
```

And to get all the information for the playlist:

```
SELECT PlObj.title as PlayListTitle, t_artist.title as
ArtistName, t_album.title as AlbumName, t_track.trackNo as
TrackNumber, TrObj.title as TrackTitle, t_genre.title as
GenreTitle, t_work.title as WorkTitle, t_composer.title as
ComposerTitle, t_track.*, t_artist.*, t_album.*, TrObj.*,
t_genre.*, t_work.*, t_composer.*, t_container.* FROM
(t_object PlObj INNER JOIN (t_container INNER JOIN (t_composer
INNER JOIN (t_work INNER JOIN (t_genre INNER JOIN (t_album
INNER JOIN (t_artist INNER JOIN t_track ON t_artist.OBJ_ID =
t_track.ARTIST_ID) ON t_album.OBJ_ID = t_track.ALBUM_ID) ON
t_genre.OBJ_ID = t_track.GENRE_ID) ON t_track.WORK_ID =
t_work.OBJ_ID) ON t_track.COMPOSER_ID = t_composer.OBJ_ID) ON
t_track.OBJ_ID = t_container.CID) ON t_container.PID =
PlObj.OBJ_ID) INNER JOIN t_object TrObj ON
t_track.OBJ_ID=TrObj.OBJ_ID ORDER BY t_container.GPID,
t_container.itemOrder
```

Just try them out in any SQLLite browser...

Now most people at this stage will think: HELP!!!!!

Ok, a little bit of help then.. I created some files for the log converter that will be good enough for most people and are a good start for others to expand their metadata export.

Make sure that your decrypted database zdb is located at the Data folder. That's where the program is looking at. Doubleclick at 01 TRACK DB INFO. If a connection can't be made, check if the zdb database ia at the correct location. If the file is connected, all fields and their aliases are shown for the provided query.

```
01 Track DB Info
------------------------------------------------

Connection   : Data Source= D:\Users\Olive Backup Exporter\Data\zdb;Version=3
Database File: D:\Users\Olive Backup Exporter\Data\zdb
SQL          : SELECT t_artist.title as ArtistName, t_album.title as
AlbumName, t_track.trackNo as TrackNumber, t_object.title as TrackTitle,
t_genre.title as GenreTitle, t_work.title as WorkTitle, t_composer.title as
ComposerTitle, t_track.*, t_artist.*, t_album.*, t_object.*, t_genre.*,
t_work.*, t_composer.* FROM (t_composer INNER JOIN (t_work INNER JOIN (t_genre
INNER JOIN (t_album INNER JOIN (t_artist INNER JOIN t_track ON t_artist.OBJ_ID
= t_track.ARTIST_ID) ON t_album.OBJ_ID = t_track.ALBUM_ID) ON t_genre.OBJ_ID =
t_track.GENRE_ID) ON t_track.WORK_ID = t_work.OBJ_ID) ON t_track.COMPOSER_ID =
t_composer.OBJ_ID) INNER JOIN t_object ON t_track.OBJ_ID=t_object.OBJ_ID ORDER
BY ArtistName, AlbumName,  TrackNumber
Value if NULL: NULL
Record Count : 14243
Field Count  : 137
------------------------------------------------
All available fields and their aliases
------------------------------------------------
0 'ArtistName' ALIAS #%FIELD0%# ALIAS #%ARTISTNAME%#
1 'AlbumName' ALIAS #%FIELD1%# ALIAS #%ALBUMNAME%#
2 'TrackNumber' ALIAS #%FIELD2%# ALIAS #%TRACKNUMBER%#
3 'TrackTitle' ALIAS #%FIELD3%# ALIAS #%TRACKTITLE%#
4 'GenreTitle' ALIAS #%FIELD4%# ALIAS #%GENRETITLE%#
5 'WorkTitle' ALIAS #%FIELD5%# ALIAS #%WORKTITLE%#
6 'ComposerTitle' ALIAS #%FIELD6%# ALIAS #%COMPOSERTITLE%#
7 'TRACK_ID' ALIAS #%FIELD7%# ALIAS #%TRACK_ID%#
8 'ARTIST_ID:1' ALIAS #%FIELD8%# ALIAS #%ARTIST_ID:1%#
9 'ALBUM_ID:1' ALIAS #%FIELD9%# ALIAS #%ALBUM_ID:1%#
10 'COMPOSER_ID' ALIAS #%FIELD10%# ALIAS #%COMPOSER_ID%#
11 'WORK_ID:1' ALIAS #%FIELD11%# ALIAS #%WORK_ID:1%#
12 'GENRE_ID:1' ALIAS #%FIELD12%# ALIAS #%GENRE_ID:1%#
```

The purpose of this file is to check a database connection and show you all available fields that can be used, to extract data.

Now doubleclick on the file "02 Track Content" It shows metadata output for the first 200 records. The following tags are shown with " <--> " as seperator

```
PATH(.\DATA\ARTIST\ALBUM\TRACKNO – TITLE) Artist Album Trackno Title Genre Year
```

If that's good enough for you, just doubleclick on "03Track Export.txt". Now all data is copied to a text file in the Data subfolder. With large databases this can take a while for both preparation and execution (12 minutes on a fast PC for a 15000 tracks size database..) Just relax, watch the counter and take a nice cup of coco...

With the MP3tag program, this file can later be imported with a format string like:

```
 %dummy% <--> %artist% <--> %album% <--> %track% <--> %title% <--> %genre% <--> %year%
```

The other files are almost the same, just with another query to get the playlist info.
The example file is a good start for the more advanced, who like to do more.

If you like Log Converter and want to use it more often or use it's other functions, I suggest that you get the latest version and manual from the download page at www.Lydo.nl

Note that the normal version does not (yet) have the SQLite functionality.

Customize the export of the database Meta tagging

If you want to change the seperator just do a find and replace on the output file via Notepad or edit the file "03 Track Export.txt" file and change the value of "[B] <--> [/B]".

Now to change the actual output, take a look at the folowing line:

[ADD2CONTENT]#A#{CHECKPATH|#@@@#%ARTISTNAME%#\%%%@@@%%%#%ALBUMN
AME%#\^#%TRACKNUMBER%# - %%%#%TRACKTITLE%##}#B##%ARTISTNAME%##B#
#%ALBUMNAME%##B##%TRACKNUMBER%##B##%TRACKTITLE%##B##%GENRETITLE%##B#
#%RECDATE%#

[/ADD2CONTENT]

- #B# are the seperators,
- The blue part in the front determines the path of the file
- [ADD2CONTENT]... [/ADD2CONTENT] adds the provide text to the content
- The remaining black parts are the fields to export.

With the "01 Track DB Info" we know the name of all 137 fields in the query. Just use them in this part and change to your own requirements.

Export the Olive hard drive

Now if the Olive did not function (e.g. white startup screen), what next? First check this excellent forum http://www.avsforum.com/forum/153-cd-players-dedicated-music-transports/1091695-official-olive-thread-opus-4-opus-6-melody-2-olive-2-olive-4-4hd-06hd.html

There are a lot of knowledge people who had their issues with Olive as well…
If the Olive will still not work and the harddisk is not broken, there is a way to get the data from the Olive.

Remove the harddisk from the unit, connect it to your computer via an USB adapter. Use the ext2fsd program to mount the last partition to a driveletter. The database is also at that location, but this time not decrypted. So copy the "zDB" file from the Olive harddisk to the target folder. Do not use the decrypt button this time, just GO!