<< Back

Pipeline

Root

Drive structure

Real drives

Drive letter	Function
C:	Windows drive
D:	Sync work drive here + other work related stuff
E:	Sync all media here (foto, music, video)

Mapped drives

Drive letter	Function
I:	Active projects:
	Audio
	Photo
	Visual
	VJ
	Web
M:	Music
P:	Files for software (settings, adons)
F:	Personal files

From the command line:

```
subst A: /d
```

Replace A with drive you want deleted

Startup

I'm executing 2 scripts on startup using a single InvisibleLauncher.vbs file referenced in “Task Scheduler”

1. CreateDrives

Using VisualSubst

2. AppsToDesktop

This is a .ps1 script, referenced to a a folder called “startup” that you can put whereever you want. In this startup, put shortcuts in the folders named after the corresponding desktop where you want them to be opened, like “Desktop1”

Work locally while keeping a mirror on NAS

Create symbolic links from your Laragon www directory to your development folders on the NAS. This keeps your working environment organized while allowing Laragon to serve the sites. `mklink /D [Link] [Target]`

Audio

Folder structure

Artwork finder

- MusicProduction
  - Practice
  - Ideas
  - InProgress
  - FinalTouches
  - Finished
  - Publish
- DJSets
- Library
  - Samples
    - AmbiguousRoyaltySounds
    - FreeUse

Configure MusicBee and Traktor for EZ playlist management

MusicBee: Ensure that MusicBee is consistently exporting the “M:\iTunes Music Library.xml” file on close. You’ve already set this up,
but double-check under MusicBee’s Preferences > Library > “export the library as an iTunes formatted XML file” is enabled and pointing to “M:\iTunes
Music Library.xml”.

Traktor: In Traktor Preferences → File Management:
Set the “iTunes/Music Library” path to “M:\iTunes Music Library.xml”. This ensures Traktor always looks at your MusicBee-exported XML.

Enable “Import Music-Folders at Startup” and add the folder(s) where your actual music files reside (e.g., M:\Music\ or wherever your tracks are

stored). This keeps the Track Collection updated with new files, though it won’t handle playlists directly.

Fire up the Python script:
It enambles synchronisation of playlists from any iTunes.xml compatible music library to any DJ software of your choice.
Here's what it does:

1. It reads an iTunes XML library file to extract tracks from user playlists
2. It copies these tracks to a designated “DJ Library” folder
3. If enabled, it converts audio files to WAV format using FFmpeg
4. It creates a new XML file with updated file paths that point to the DJ Library folder

Key features:

- Filters tracks to only include supported audio formats (.mp3, .flac, .wav, etc.)
- Skips system and auto-generated playlists
- Can skip files that already exist in the destination folder
- Provides detailed analysis of missing tracks for debugging
- Updates file locations in the XML for compatibility with DJ software

The script runs through three main steps:

1. Analyzing the iTunes library to extract playlist tracks
2. Copying and converting tracks to the DJ Library folder
3. Creating an updated XML file with the new file paths

This allows a you to maintain a playlists in iTunes/Musicbee while having those same tracks available in any DJ software in the preferred format.

Photo

Folder structure

- Year
  - 01_Incoming
  - 02_Outgoing

File structure

Template: YYYY-MM-DD_Location_Subject(abbreviated, PubLib (Public Library))_FileVersion.Extention
Example: 2020-12-19_Zele_PubLib_001.jpg

Visual

Folder structure

- YYYY-MM-DD_NameClient_NameProject
  - _Library
  - 00_Incoming
    - 01_Preproduction
    - 02_Production
  - 01_Preproduction
    - 01_Moodboard
    - 02_Storyboard
  - 02_Production
    - _Blender
    - _DaVinci
    - _Houdini
    - Scene_01
  - 03_Outgoing
    - YYYY-MM-DD

File structure

Naming Convention

CamelCaps, geen spaties, underscores/streepjes enkel gebruiken bij opdeling filenaam.
Maak genoeg iteraties van je bestand, zodat je steeds een versie terug kan gaan, moest er iets mislopen.

Models

Export files (FBX, OBJ) hebben steeds zelfde versienummer als work file (Blender, Maya).
Afkortingen:

- Character: CH
- Prop: P
- Foilage: F

Template: Project_SceneNumber_Modeltype-ModelName_FileVersion.Extension
Example: Silver_000_CH-Nora_v001.blend

Texture Maps

Export files (PNG, JPG) hebben steeds zelfde versienummer als work files (Substance, Photoshop)
Afkortingen:

- Diffuse: DIFF
- Ambient Oclusion: AO
- Glossines: G

- Normal: N
- Roughness: R
- Subsurface Scattering: SSS
- Metallic: M

Template: Project_SceneNumber_TextureName_FileVersion_TextureType.Extension
Example: Silver_010_NoraHair_v001_DIFF.PNG

Scenes

The count of scenes starts from 010, models with the scene number 000 are not restricted to one scene.

Template: Project_SceneNumber-Description_FileVersion.Extension
Example: Silver_010-MudWorld_v001.blend

In 3D Software

Max 5 letters voor afkortingen
Afkortingen:

- Group: GRP
- Mesh: MESH
- Curve: CRVE
- Joint: JNT
- Light: LGHT
- Camera: CAM
- Locator: LOC
- Controller: CTRL
- Low Poly: LP
- High Poly: HP
- Forces: FRCE

Template: Type_Name_PolyType
Example: GRP_Nora_LP

Reference Library

For my collection of reference images, I only use broad categories as folders, the other information I put in the name and metadata.

I'm using a combination of tools, including Advanced Renamer, digiKam, Python and a tool from GitHub called taggui.

Workflow

First, I will change the name with Advanced Renamer.
Using TagGUI, I will then let the program generate captions into seperate text files, wich are named after the image they are describing.
Now we need the Python script to take the information from the text file, and add it to the metadata of the images.
For clean organisation, I opted to convert all files to .jpg. This to make sure the metadata is displayed in a clean way.

The name

CATEGORY_SUBJECT_inc Nr

The metadata

A description/caption in the images “Title” and “Subject” fields.

Sources

https://www.youtube.com/watch?v=6KCtPnam6Sk&ab_channel=TimvanHelsdingen
https://github.com/alexanderrichtertd/plex/wiki/Project-Structure
https://www.youtube.com/watch?v=o2LqKH6ahDA

Pipeline @ Blender
PSVirtualDesktop - commandlets to manage virtual desktops