<< Back

Structures

Drives

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
M:	Music
P:	Personal files for software (settings, adons)

From the command line, subst [drive letter]: /d deletes the drive ===== *Music* ===== *Artwork finder* ==== *Folder structure* ==== *MusicProduction *Practice *Ideas *InProgress *FinalTouches *Finished *Publish *DJSets *Library *Samples *AmbiguousRoyaltySounds *FreeUse ===== *Photography* ===== ==== *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
*===== CG ===== ==== 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. == The script == <code Python [enable_line_numbers=“true”]> import os from PIL import Image as PilImage, PngImagePlugin from pyexiv2 import Image as Pyexiv2Image def add_text_to_image_metadata(image_folder): try: # Iterate through all files in the specified folder for filename in os.listdir(image_folder): image_path = os.path.join(image_folder, filename) text_file_path = os.path.join(image_folder, f“{os.path.splitext(filename)[0]}.txt”) print(f“Processing {image_path}”) # Debugging line if os.path.exists(text_file_path): # Read the content from the text file with open(text_file_path, 'r') as file: caption_text = file.read() # Convert filename to lowercase for extension checking filename_lower = filename.lower() if filename_lower.endswith[1]: # Load the image's metadata image = Pyexiv2Image(image_path) # Add the caption to the image's EXIF UserComment and ImageDescription fields image.modify_exif({'Exif.Photo.UserComment': caption_text, 'Exif.Image.ImageDescription': caption_text}) elif filename_lower.endswith('.png'): # Convert the PNG image to JPEG image = PilImage.open(image_path) rgb_image = image.convert('RGB') rgb_image.save(os.path.splitext(image_path)[0] + '.jpg', 'JPEG') # Load the image's metadata image = Pyexiv2Image(os.path.splitext(image_path)[0] + '.jpg') # Add the caption to the image's EXIF UserComment and ImageDescription fields image.modify_exif({'Exif.Photo.UserComment': caption_text, 'Exif.Image.ImageDescription': caption_text}) # Delete the original PNG image os.remove(image_path) # Delete the text file os.remove(text_file_path) print(“Processing completed for all images.”) except Exception as e: print(f“Error: {e}”) # Example usage: Specify the folder containing your images image_folder_path = '/path/to/your/source/folder' add_text_to_image_metadata(image_folder_path) }(); </code>
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

1)
'.jpg', '.jpeg'

From:
http://floriandheer.com/wiki/ - **Brain II**

Permanent link:
http://floriandheer.com/wiki/doku.php?id=start:structures&rev=1722415676

Last update: **2024/07/31 10:47**