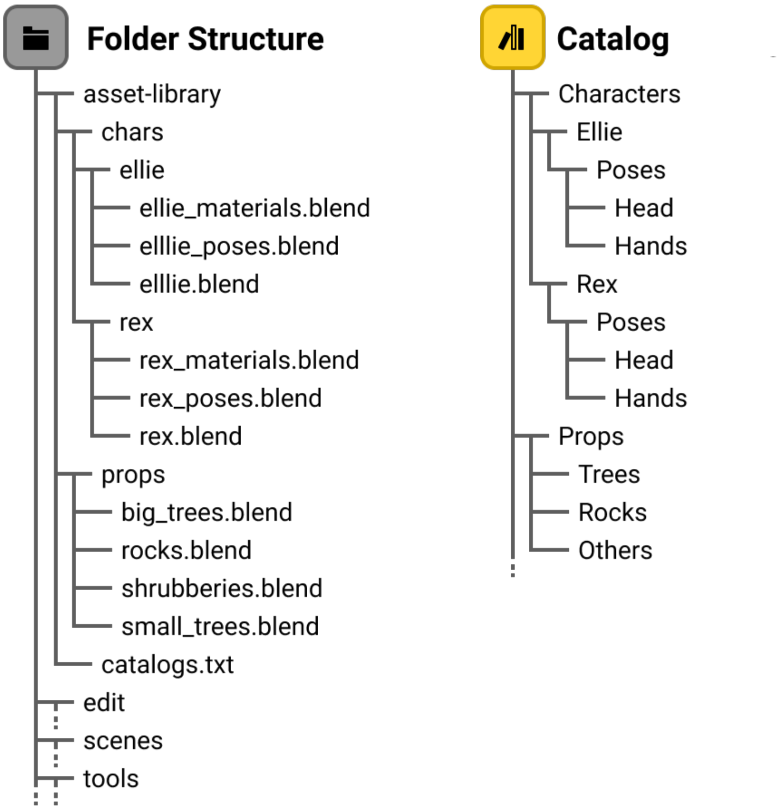
Skip to content

Asset Catalogs

Asset Catalogs help you to organize your assets. They look a little bit like file directories, but they are completely independent of the location of your blend-files. Assign each asset in a blend-file to its own catalog, or have one big catalog with all the assets of all the blend-files combined. It's all up to yo

Similar to Collections, catalogs can be nested i.e. you can have a main catalog that contains several nested catalogs. For example, this allows you to hav catalog of assets for "Furniture" with sub-catalogs of "Tables", "Chairs", "Lamps", etc…

For more technical information, see Asset Catalogs on the Blender Developer Documentation.



Example file system and catalog structures.

The Home Location of Assets

There can be as many catalogs as you want, but **an asset can be assigned to a single catalog** at a time. This is similar to a file system, where a file is only in one directory (ignoring advanced things like symbolic links).

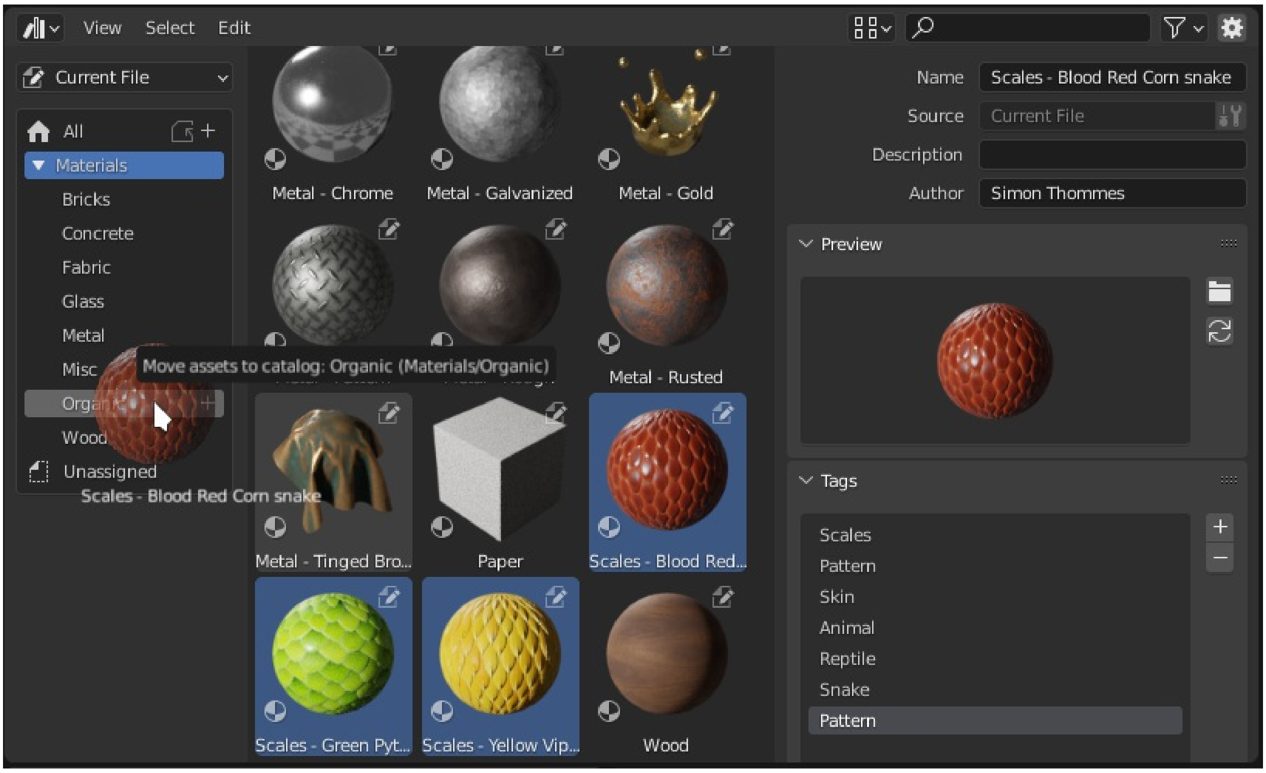
Catalogs themselves can be nested and moved by dragging and dropping. Moving a catalog will not modify the assets it contains; they will simply move along to the new location of the catalog.

Selecting a catalog in the Asset Browser will show all assets in that catalog and in child catalogs. So, in the preceding example, selecting `Characters/Ellie/Poses` will also show assets from `Characters/Ellie/Poses/Head` and `Characters/Ellie/Poses/Hands`.

Creating Catalogs

New catalogs can be created in the Asset Browser through Header ‣ Catalog ‣ New Asset Catalog. Once the catalog is created you can double `LMB` on it's name in the *Source List* region of the editor to give the catalog a more descriptive name. Catalogs can also be created in this region by clicking th plus icon found at the top of the tree view.

Assigning an Asset



Move assets to catalog: Organic (Materials/Organic)

Name: Scales - Blood Red Corn snake
Source: Current File
Description:
Author: Simon Thommes

Tags: Scales, Pattern, Skin, Animal, Reptile, Snake, Pattern

Assigning a selection of "Scale material" assets to a catalog.

To assign assets to a catalog, just select and drag the assets on top of the catalog.

Tip

You can assign an asset to the "Unassigned" catalog, this will remove it from any existing catalogs.

Saving Catalogs

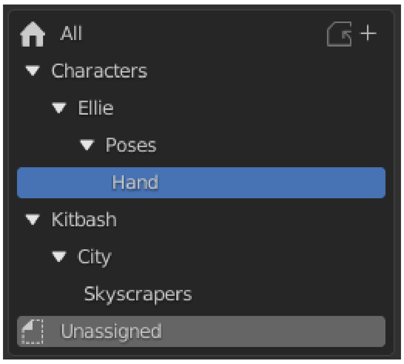
Saving catalogs makes any edits to any catalogs permanent by writing the current set up to the asset library. Catalogs can be saved in the Asset Browser through Header ‣ Catalog ‣ Save Asset Catalog. Once the catalog is created you can double `LMB` on it's name Catalogs can also be saved in the *Source List* region of the editor by clicking the save icon found at the top of the tree view.

Components of a Catalog

Each catalog consists of a *catalog path*, a *UUID*, and a *simple name*. Normally you would only deal with the catalog path; the rest is for internal Blend use and/or for emergency situations.

Catalog Path

The path of a catalog determines where in the catalog hierarchy the catalog is shown. Examples are `Characters/Ellie/Poses/Hand` or `Kitbash/City/Skyscrapers`, which would result in the following catalog tree. The highlighted catalog has path `Characters/Ellie/Poses/Hand`.



Example tree of asset catalogs

Example tree of asset catalogs.

UUID

Each catalog has a UUID, which is normally hidden from the user interface (enable Developer Extras and the experimental Asset Debug Info option to s them). This is what is stored in the asset, and what determines the “identity” of the catalog. As a result, a catalog can be renamed or moved around (i.e. you can change its path), and all assets that are contained in it will move along with it. This only requires a change to the catalog itself, and not to any asse blend-file.

Simple Name

Each catalog has an optional *simple name*. This name is stored along with the UUID in each asset. The purpose is to make it possible for humans to recognize the catalog the asset was assigned to, even when the *catalog definition file* (see below) is lost.

Like the UUID, the simple name is normally hidden from the user interface. Enable Developer Extras in the interface preferences to make it visible in the Asset Browser.

Catalog Definition Files

Asset catalogs are stored in Catalog Definition Files (CDFs). Blender 3.0 supports a single CDF per asset library. It is stored in `blender_assets.cats.txt` in the root directory of the asset library. If the file does not exist, Blender will create it when the catalogs are save When catalogs are changed, Blender updates that file, but also creates a backup of the previous state to a file named `blender_assets.cats.txt~`.

Which File to Write To

Asset catalogs can be saved independently of the blend-file; the catalog editor has its own “Save” button.

Format

Catalog Definition Files (CDFs) are relatively simple text files, encoded in UTF-8. Each CDF consists of a version indicator, and a line of text per catalo Each catalog line is colon-separated, of the form `{UUID}:{path}:{simple name}`.

Example

This is an example of a valid catalog definition file:

```
# This is an Asset Catalog Definition file for Blender.
#
# Empty lines and lines starting with `#` will be ignored.
# The first non-ignored line should be the version indicator.
# Subsequent lines are of the format "CATALOG_UUID:catalog/path/for/assets:simple catalog

VERSION 1

313ea471-7c81-4de6-af81-fb04c3535d0e:catalog/without/simple/name:
ee9c7b60-02f1-4058-bed6-539b8d2a6d34:character/Ellie/poselib:character-Ellie-poselib
cd66bf52-58f4-45cb-a4e2-dc0e0ee8f3fe:character/Ellie/poselib:character-Ellie
4eb44ec6-3424-405b-9782-ca006953e799:character/Ellie/poselib/white space:character-Ellie-p
b63ed357-2511-4b96-8728-1b5a7093824c:character/Ružena/poselib:Ružena pose library
dcdee4df-926e-4d72-b995-33106983bb9a:character/Ružena/poselib/face:Ružena face
fb698f2e-9e2b-4146-a539-3af292d44899:character/Ružena/poselib/hand:Ružena hands
```

Valid Catalog Paths

Catalog paths follow the following rules:

- All paths are absolute; there is no difference between `/a/b` and `a/b`.
- Only `/` as separator (no `\`; think less filesystem path and more URL).
- Not empty (it's required for a valid catalog).
- No empty components (so not `a//b`; `a/b` is fine).
- Invalid characters: `:`, `\`.
- Paths are always interpreted as UTF-8.