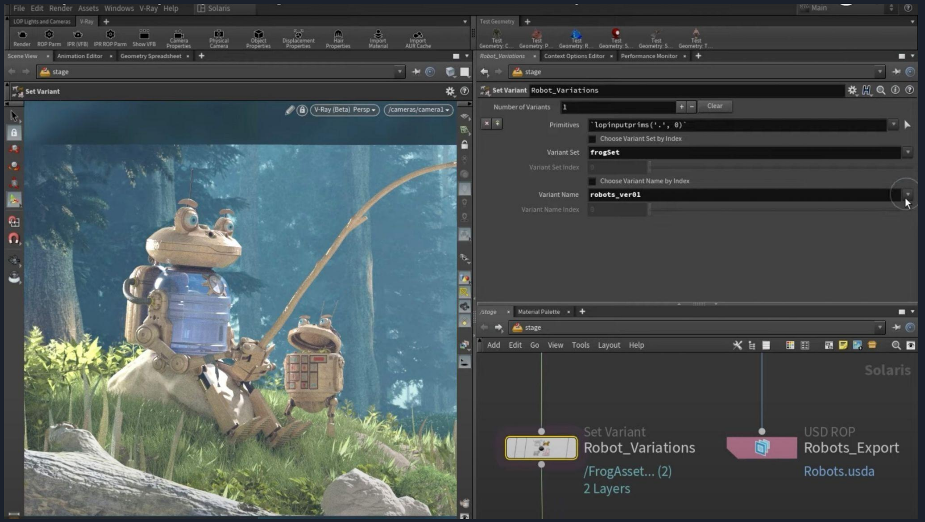
Houdini Solaris Integration

Team members: Catherine Cheng, Jackie Li, Yifan Lu, Michael Mason, Annie Qiu, Rain Yan, Linda Zhu, Crystal Zou

What is Houdini Solaris?

Houdini Solaris Introduction Speedrun

- Houdini native integration with USD
- According to SideFX:
    - “Suite of look-development, layout, and lighting tools that empower artists to create USD-based scene graphs”
- Can access a variety of rendering engines!


Set Variant Robot_Variations

Number of Variants 1

Primitives `lopinputprims('.', 0)`

□ Choose Variant Set by Index

Variant Set frogSet

Variant Set Index 0

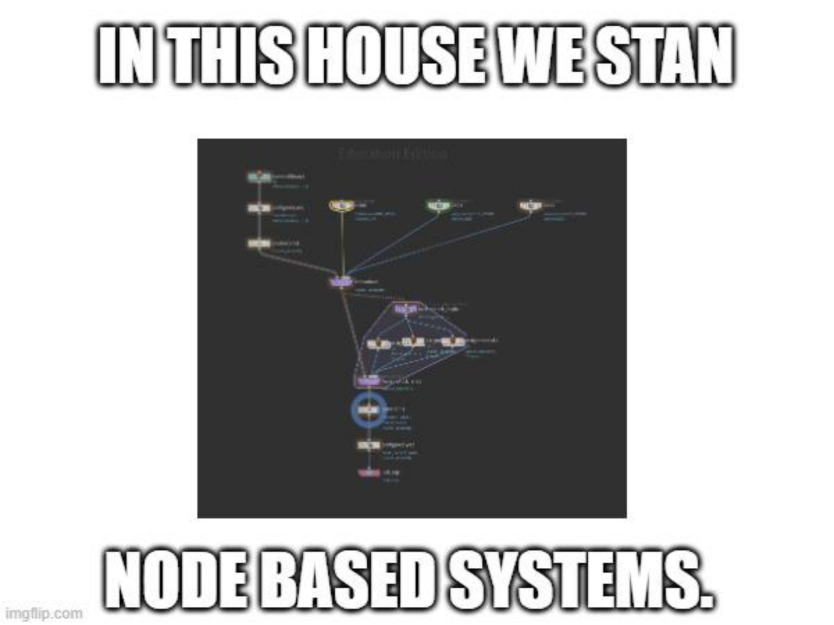
□ Choose Variant Name by Index

Variant Name robots_ver01

Variant Name Index 0

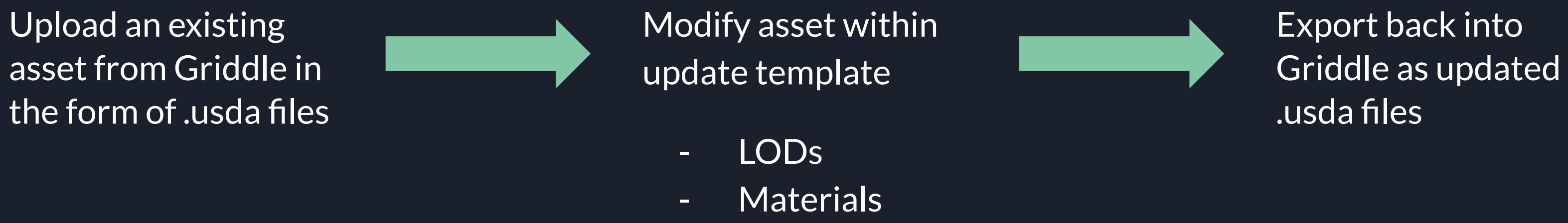
Why Solaris?

- Would be useful to have a native integration to and from Houdini!
- B/c Houdini is becoming more popular to use as a tool
- Can modify asset structure with a visual, node-based system



Workflow

Basic Overview of Workflow: Updating Asset



Upload an existing asset from Griddle in the form of .usda files

Modify asset within update template

- LODs
- Materials

Export back into Griddle as updated .usda files

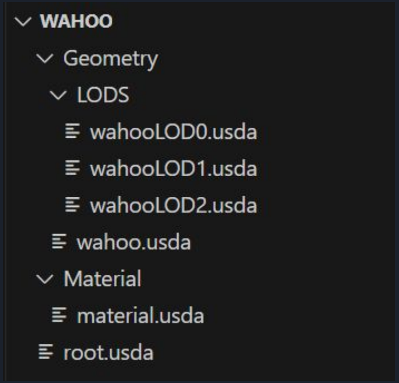
Basic Overview of Workflow: Create New Asset

Create new asset in local Houdini → From local source, upload asset into LOD references → Add or modify any material variants → Export into Griddle as a new asset with .usda files

New Component Asset Structure

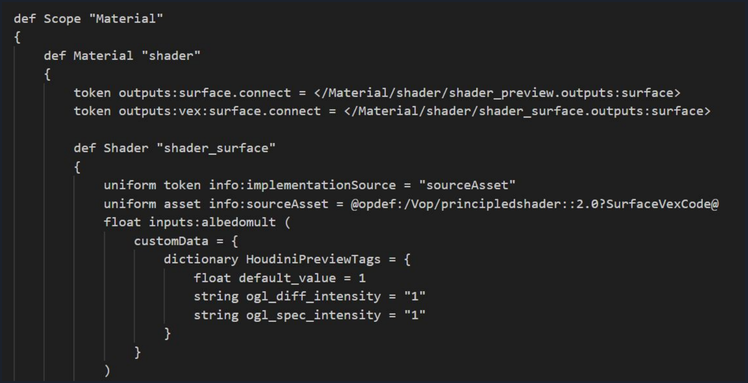
New Component Asset Structure

Saving Hierarchy



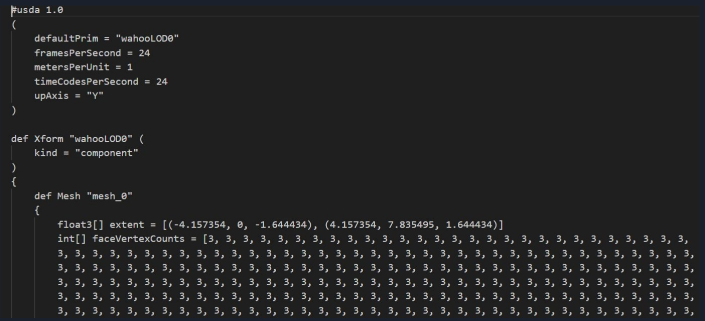
```
WAHOO
  Geometry
    LODS
      wahooLOD0.usda
      wahooLOD1.usda
      wahooLOD2.usda
    wahoo.usda
  Material
    material.usda
  root.usda
```

material.usda



```
def Scope "Material"
{
    def Material "shader"
    {
        token outputs:surface.connect = </Material/shader/shader_preview.outputs:surface>
        token outputs:vex:surface.connect = </Material/shader/shader_surface.outputs:surface>

        def Shader "shader_surface"
        {
            uniform token info:implementationSource = "sourceAsset"
            uniform asset info:sourceAsset = @opdef:/Vop/principledshader::2.0?SurfaceVexCode@
            float inputs:albedomult (
                customData = {
                    dictionary HoudiniPreviewTags = {
                        float default_value = 1
                        string ogl_diff_intensity = "1"
                        string ogl_spec_intensity = "1"
                    }
                }
            )
```

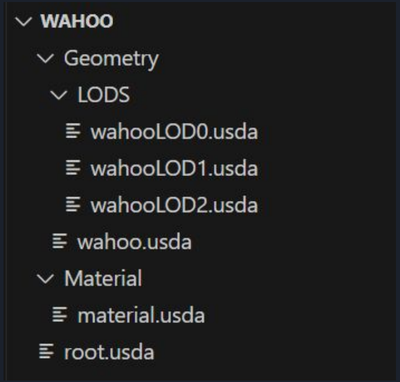
wahooLOD#.usda



```
#usda 1.0
(
    defaultPrim = "wahooLOD0"
    framesPerSecond = 24
    metersPerUnit = 1
    timeCodesPerSecond = 24
    upAxis = "Y"
)

def Xform "wahooLOD0" (
    kind = "component"
)
{
    def Mesh "mesh_0"
    {
        float3[] extent = [(-4.157354, 0, -1.644434), (4.157354, 7.835495, 1.644434)]
        int[] faceVertexCounts = [3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
        3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3,
```

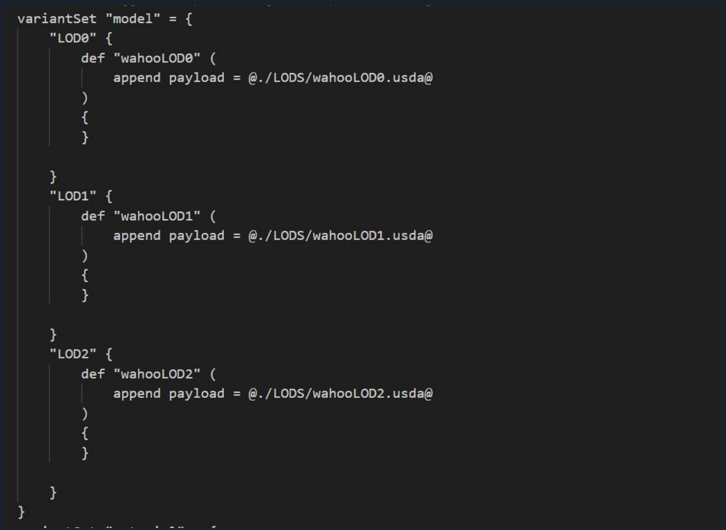
New Component Asset Structure

Saving Hierarchy



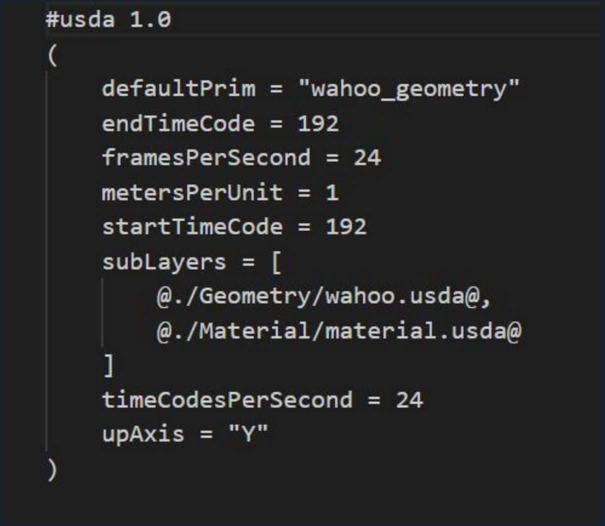
- WAHOO
  - Geometry
    - LODS
      - wahooLOD0.usda
      - wahooLOD1.usda
      - wahooLOD2.usda
    - wahoo.usda
  - Material
    - material.usda
  - root.usda

wahoo.usda



```
variantSet "model" = {
    "LOD0" {
        def "wahooLOD0" (
            append payload = @./LODS/wahooLOD0.usda@
        )
        {
        }

    }
    "LOD1" {
        def "wahooLOD1" (
            append payload = @./LODS/wahooLOD1.usda@
        )
        {
        }

    }
    "LOD2" {
        def "wahooLOD2" (
            append payload = @./LODS/wahooLOD2.usda@
        )
        {
        }

    }
}
```

root.usda



```
#usda 1.0
(
    defaultPrim = "wahoo_geometry"
    endTimeCode = 192
    framesPerSecond = 24
    metersPerUnit = 1
    startTimeCode = 192
    subLayers = [
        @./Geometry/wahoo.usda@,
        @./Material/material.usda@
    ]
    timeCodesPerSecond = 24
    upAxis = "Y"
)
```

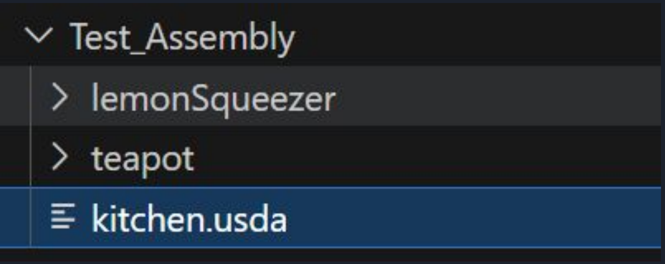
New Component Asset Structure

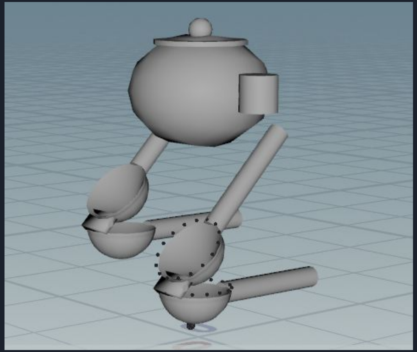
- All in usda format!

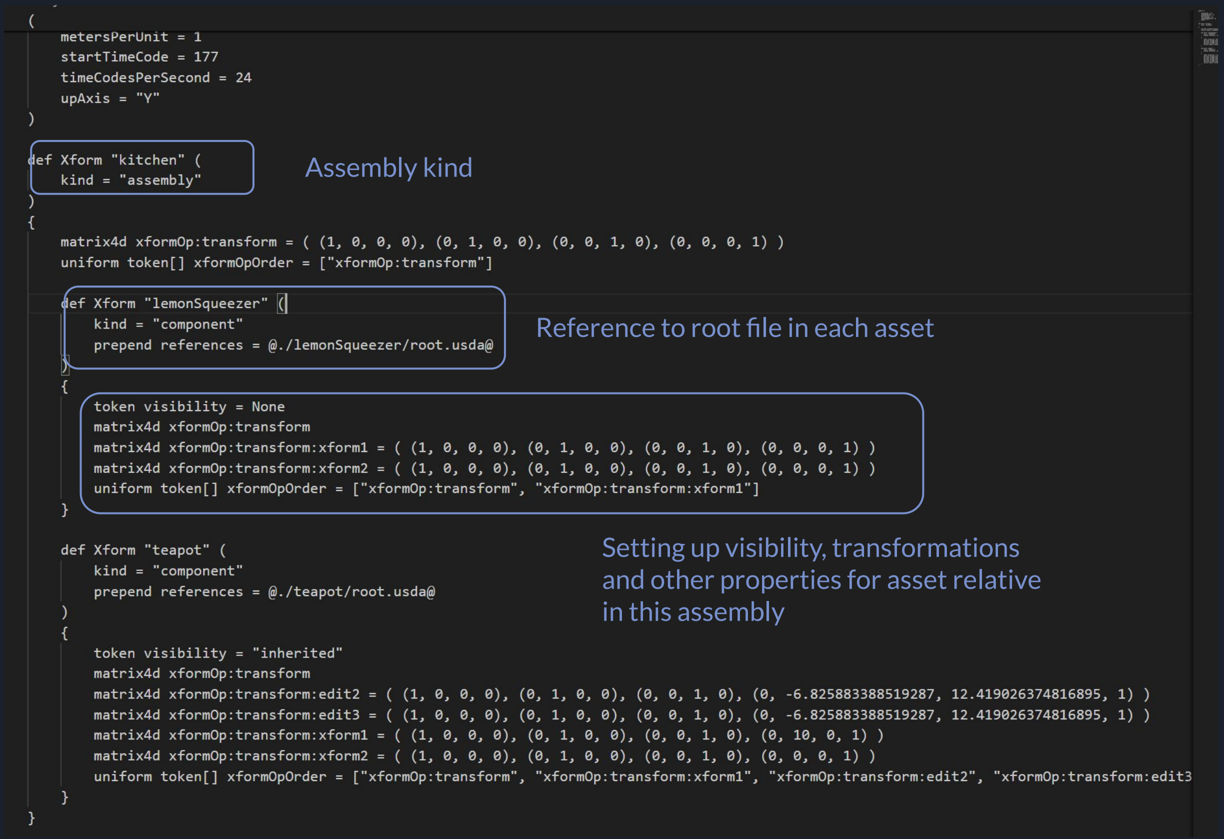
New Assembly Asset Structure

New Assembly Asset Structure

Saving Hierarchy







```
(
    metersPerUnit = 1
    startTimeCode = 177
    timeCodesPerSecond = 24
    upAxis = "Y"
)

def Xform "kitchen" (
    kind = "assembly"
)
{
    matrix4d xformOp:transform = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1) )
    uniform token[] xformOpOrder = ["xformOp:transform"]

    def Xform "lemonSqueezer" (
        kind = "component"
        prepend references = @./lemonSqueezer/root.usda@
    )
    {
        token visibility = None
        matrix4d xformOp:transform
        matrix4d xformOp:transform:xform1 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1) )
        matrix4d xformOp:transform:xform2 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1) )
        uniform token[] xformOpOrder = ["xformOp:transform", "xformOp:transform:xform1"]
    }

    def Xform "teapot" (
        kind = "component"
        prepend references = @./teapot/root.usda@
    )
    {
        token visibility = "inherited"
        matrix4d xformOp:transform
        matrix4d xformOp:transform:edit2 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, -6.825883388519287, 12.419026374816895, 1) )
        matrix4d xformOp:transform:edit3 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, -6.825883388519287, 12.419026374816895, 1) )
        matrix4d xformOp:transform:xform1 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 10, 0, 1) )
        matrix4d xformOp:transform:xform2 = ( (1, 0, 0, 0), (0, 1, 0, 0), (0, 0, 1, 0), (0, 0, 0, 1) )
        uniform token[] xformOpOrder = ["xformOp:transform", "xformOp:transform:xform1", "xformOp:transform:edit2", "xformOp:transform:edit3
    }
}
```

Assembly kind

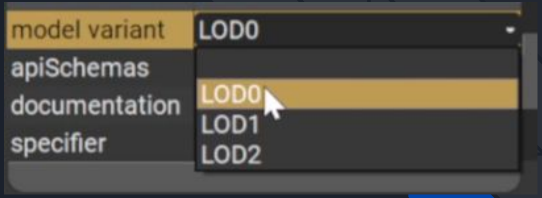
Reference to root file in each asset

Setting up visibility, transformations and other properties for asset relative in this assembly

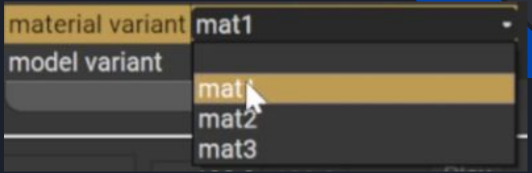
Create New Asset/Assembly

Create New Component Asset

- What users can do?
  - Create own models or import the existing obj files
  - Create new materials or change default shader settings
- What do they get?
  - Output new Component Asset with Variants
    - Geometry Set
    - Material Set



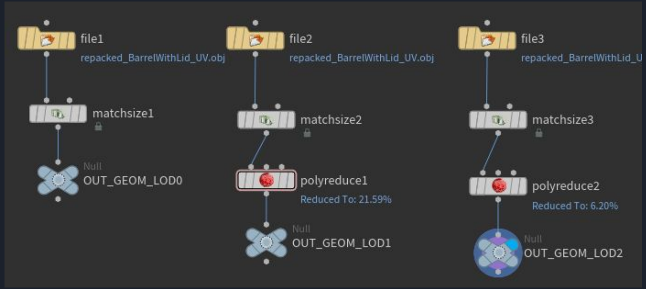
model variant LOD0

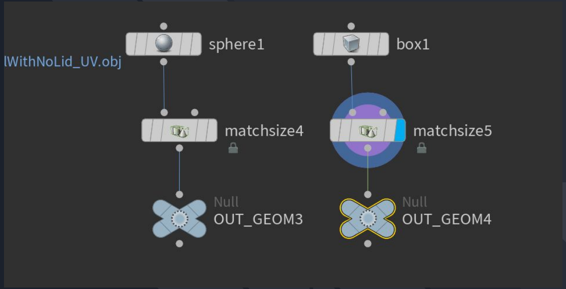


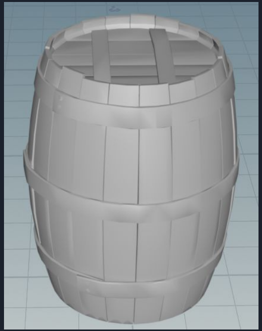
material variant mat1

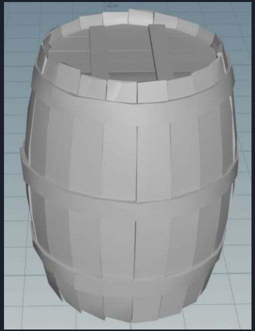
Create New Component Asset

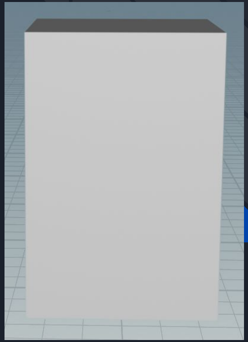
- Set models
  - Import obj or create models





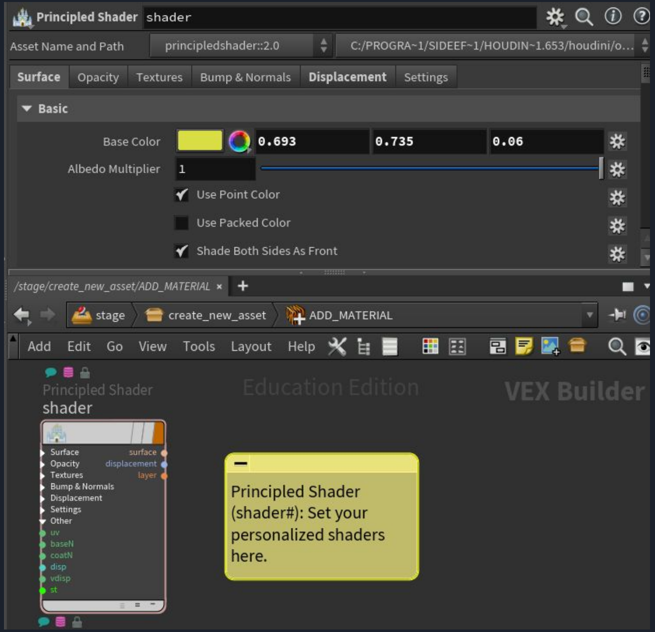






Create New Component Asset

- Set Materials
  - Import texture or create own



Principled Shader shader

Asset Name and Path: principledshader::2.0 — C:/PROGRA~1/SIDEEF~1/HOUDIN~1.653/houdini/o...

Surface | Opacity | Textures | Bump & Normals | Displacement | Settings

Basic

Base Color	0.693 0.735 0.06
Albedo Multiplier	1

☑ Use Point Color

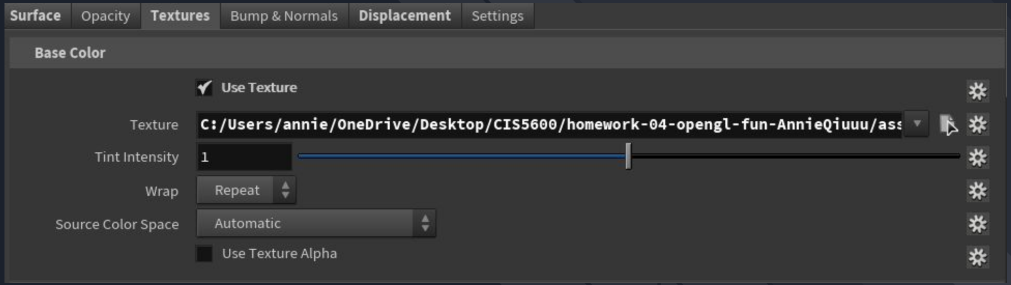
□ Use Packed Color

☑ Shade Both Sides As Front

/stage/create_new_asset/ADD_MATERIAL

stage create_new_asset ADD_MATERIAL

Principled Shader (shader#): Set your personalized shaders here.



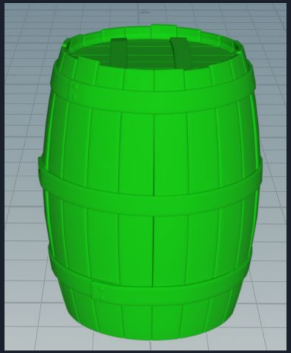
Surface | Opacity | Textures | Bump & Normals | Displacement | Settings

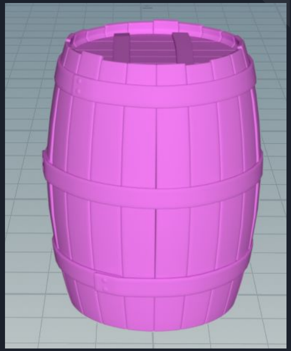
Base Color

☑ Use Texture

Texture	C:/Users/annie/OneDrive/Desktop/CIS5600/homework-04-opengl-fun-AnnieQiuuu/ass
Tint Intensity	1
Wrap	Repeat
Source Color Space	Automatic

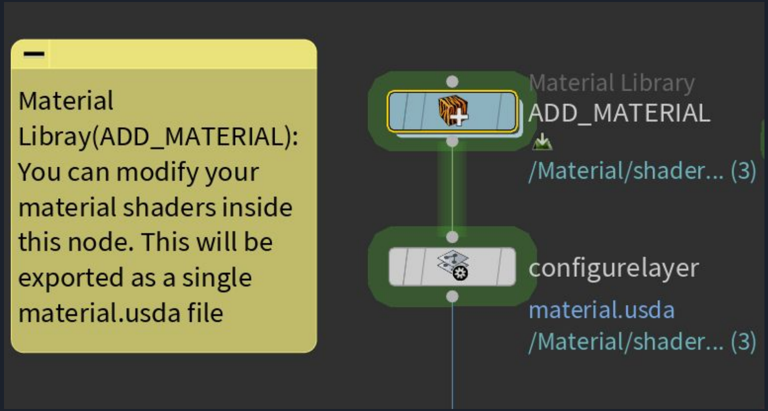
□ Use Texture Alpha





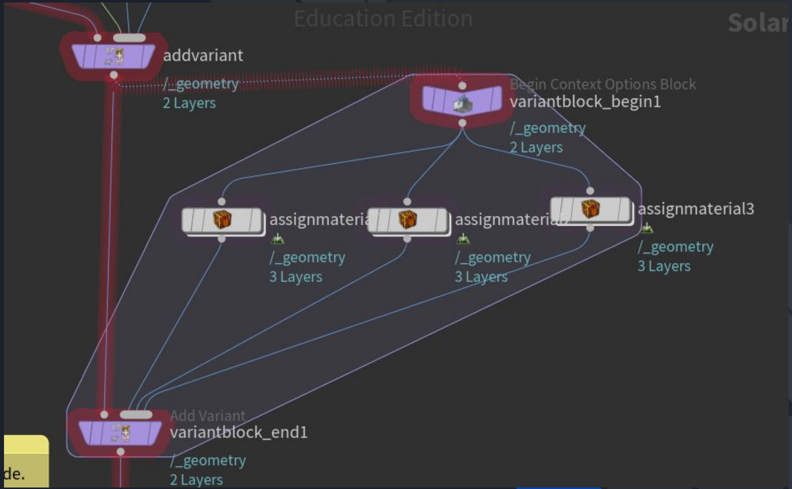
Create New Component Asset

Create Materials



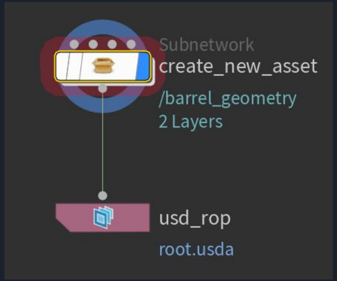
Material Libray(ADD_MATERIAL): You can modify your material shaders inside this node. This will be exported as a single material.usda file

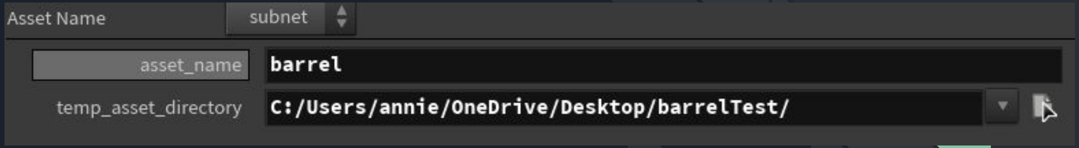
Assign Materials



Create New Component Asset

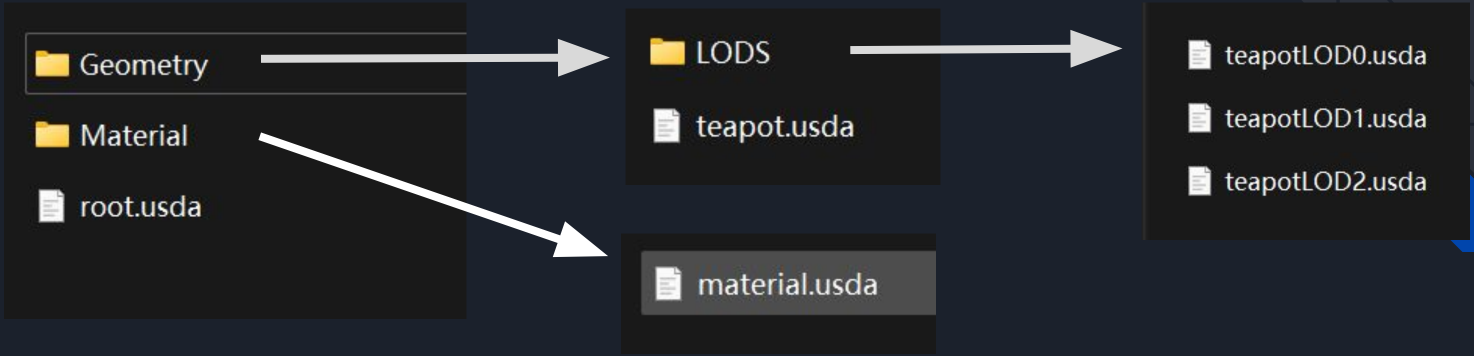
- Input Name of Asset and file path





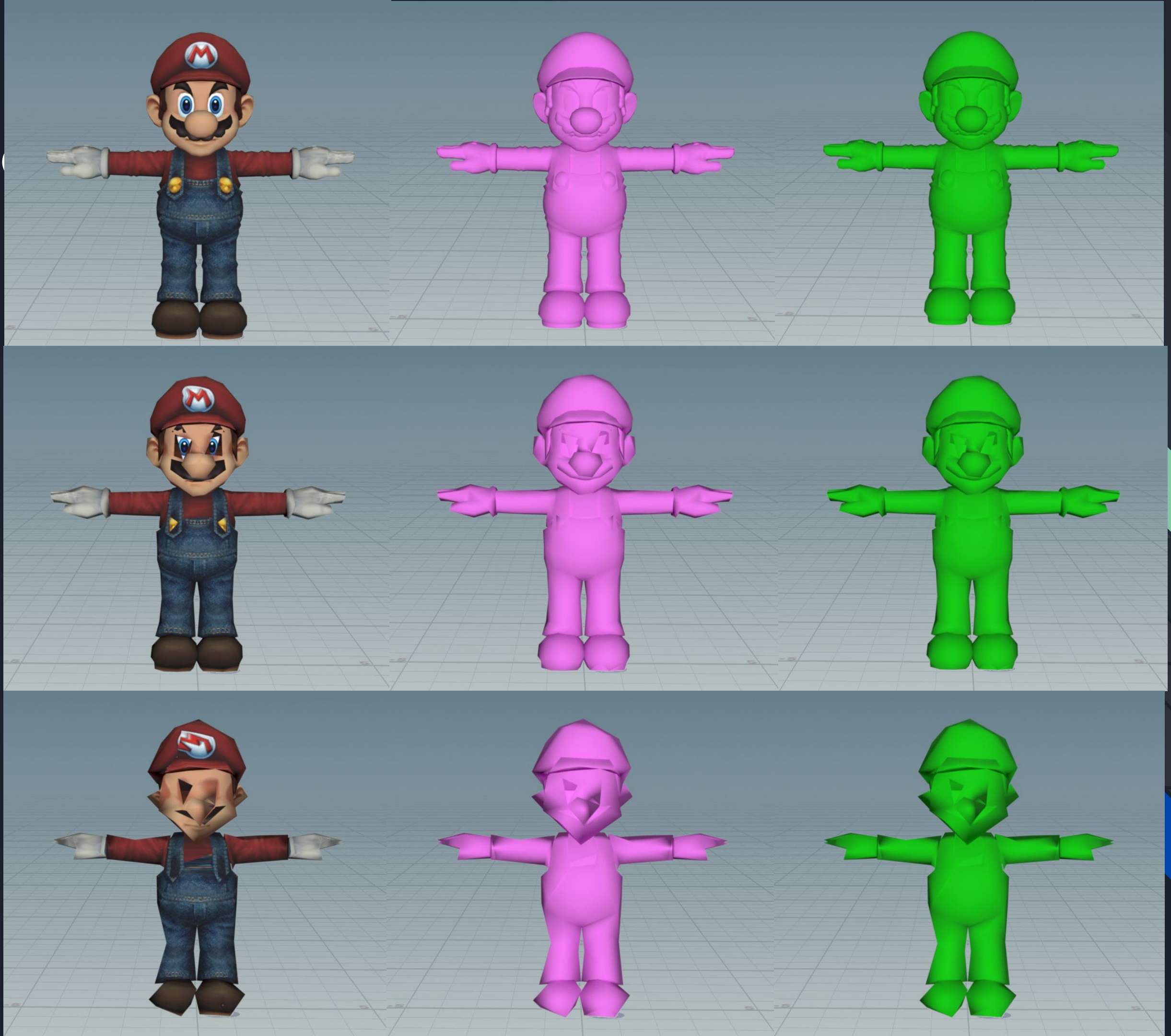
asset_name: barrel

temp_asset_directory: C:/Users/annie/OneDrive/Desktop/barrelTest/



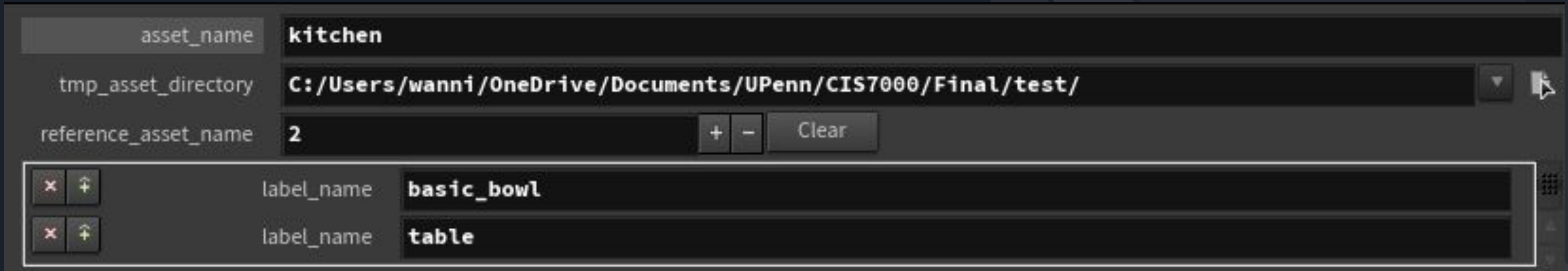
Creat

- What



Create New Assembly

- Create and export component assets



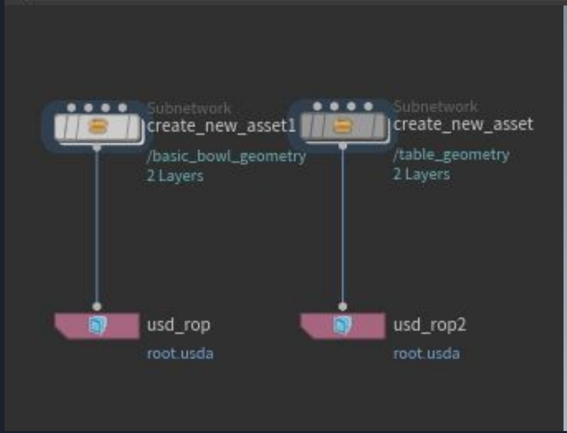
asset_name: kitchen

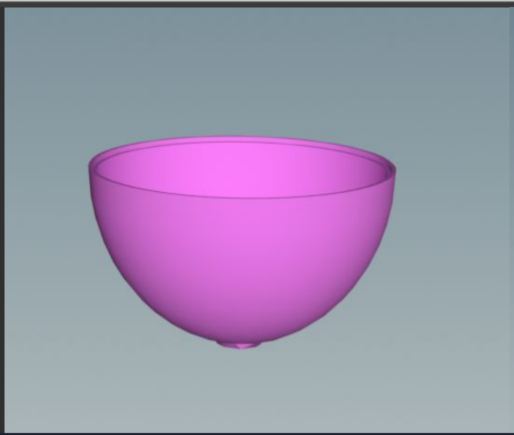
tmp_asset_directory: C:/Users/wanni/OneDrive/Documents/UPenn/CIS7000/Final/test/

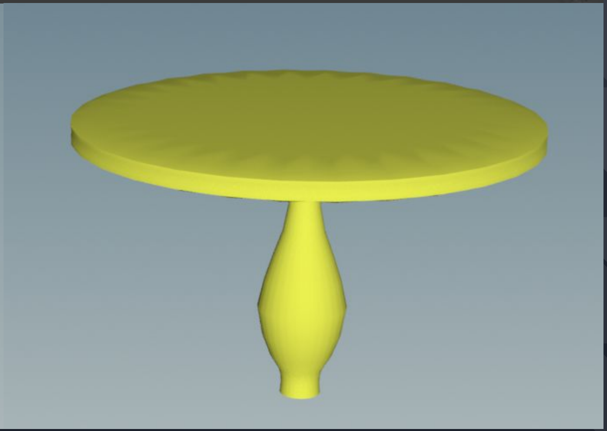
reference_asset_name: 2 + − Clear

label_name: basic_bowl

label_name: table

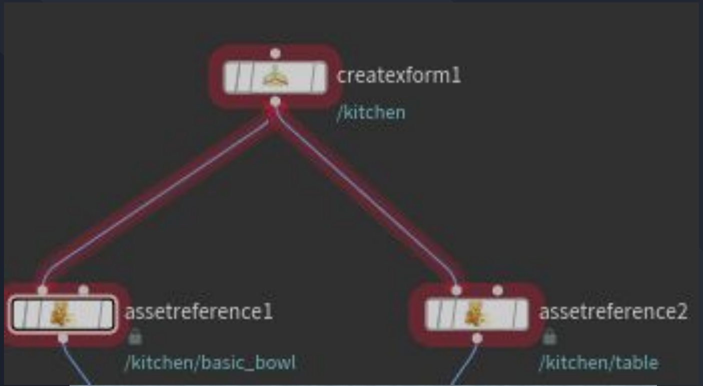


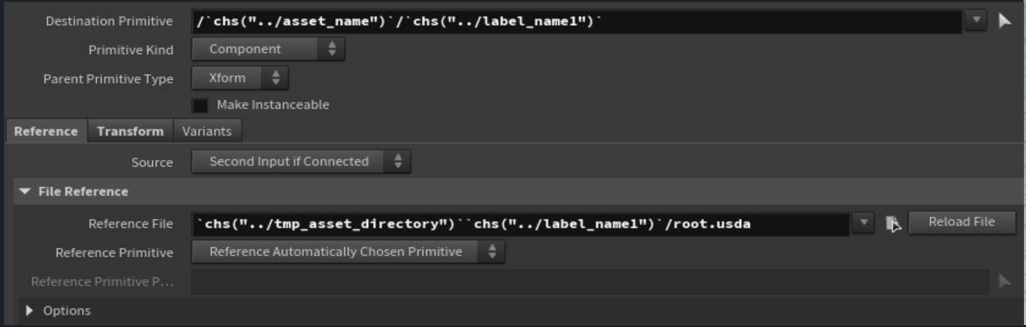




Create New Assembly

- Create asset references nodes
    - Set the hierarchy of the reference asset
    - Adjust the transformation of the component assets





Destination Primitive: /`chs("../asset_name")`/`chs("../label_name1")`

Primitive Kind: Component

Parent Primitive Type: Xform

□ Make Instanceable

Source: Second Input if Connected

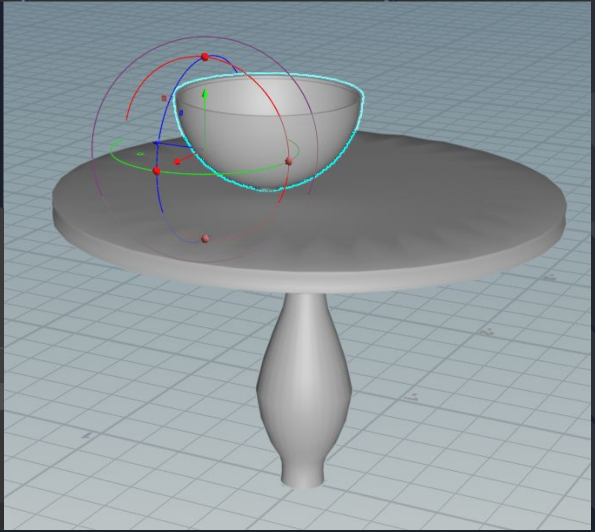
▼ File Reference

Reference File: `chs("../tmp_asset_directory")``chs("../label_name1")`/root.usda

Reference Primitive: Reference Automatically Chosen Primitive

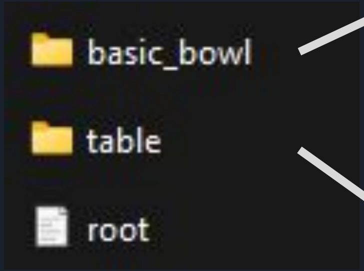
Reference Primitive P...

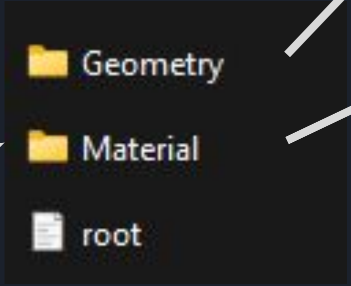
▶ Options

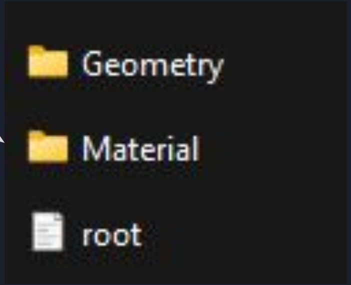


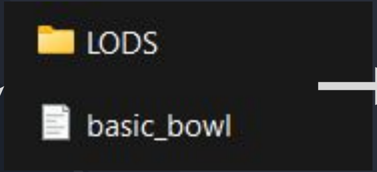
Create New Assembly

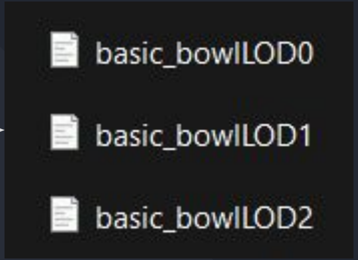
- Merge and export assembly assets
- File structure

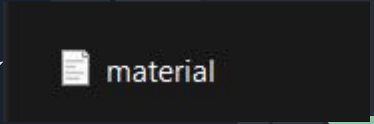

basic_bowl
table
root

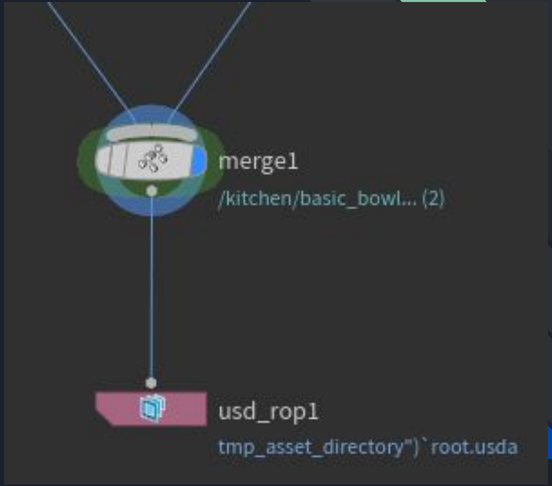

Geometry
Material
root


Geometry
Material
root


LODS
basic_bowl


basic_bowlLOD0
basic_bowlLOD1
basic_bowlLOD2


material

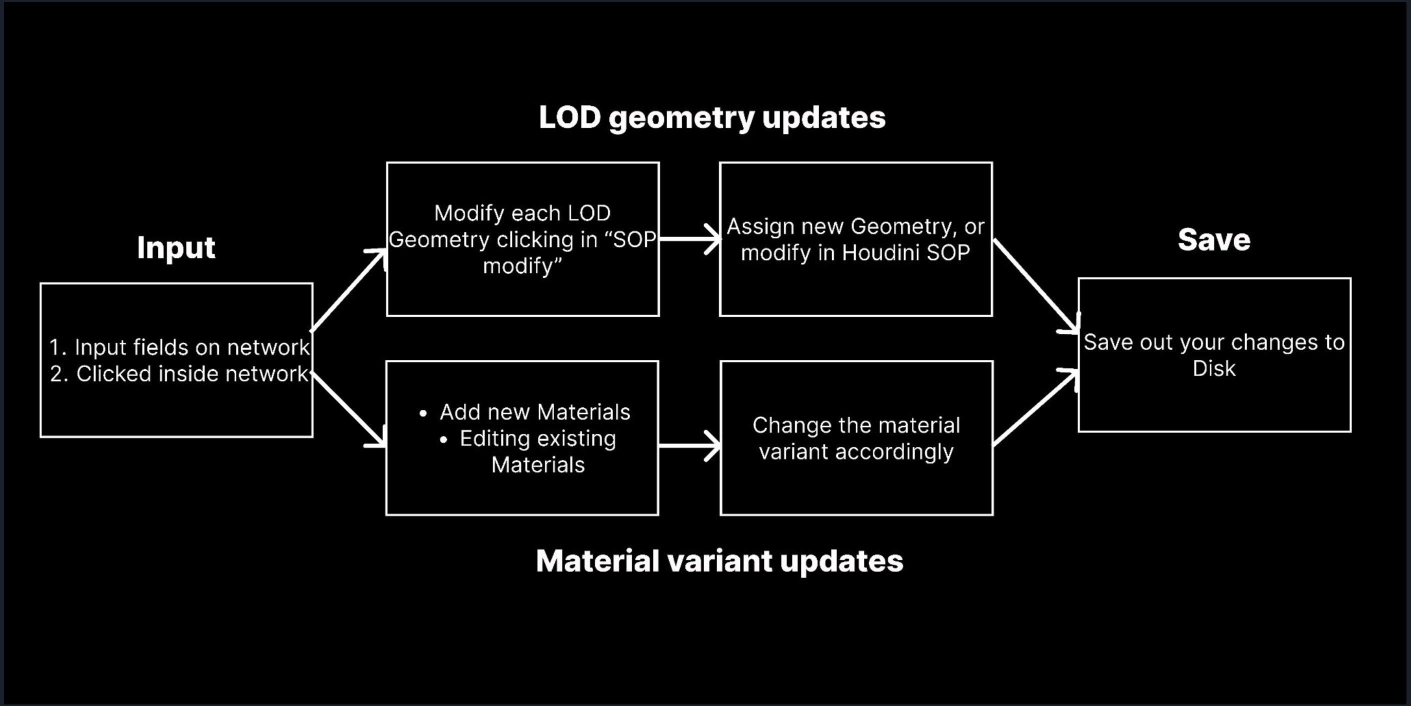

merge1
/kitchen/basic_bowl... (2)
usd_rop1
tmp_asset_directory")`root.usda

Update Asset/Assembly

Update new asset

Before

After



Input

1. Input fields on network
2. Clicked inside network

LOD geometry updates

Modify each LOD Geometry clicking in "SOP modify"

Assign new Geometry, or modify in Houdini SOP

Material variant updates

- Add new Materials
- Editing existing Materials

Change the material variant accordingly

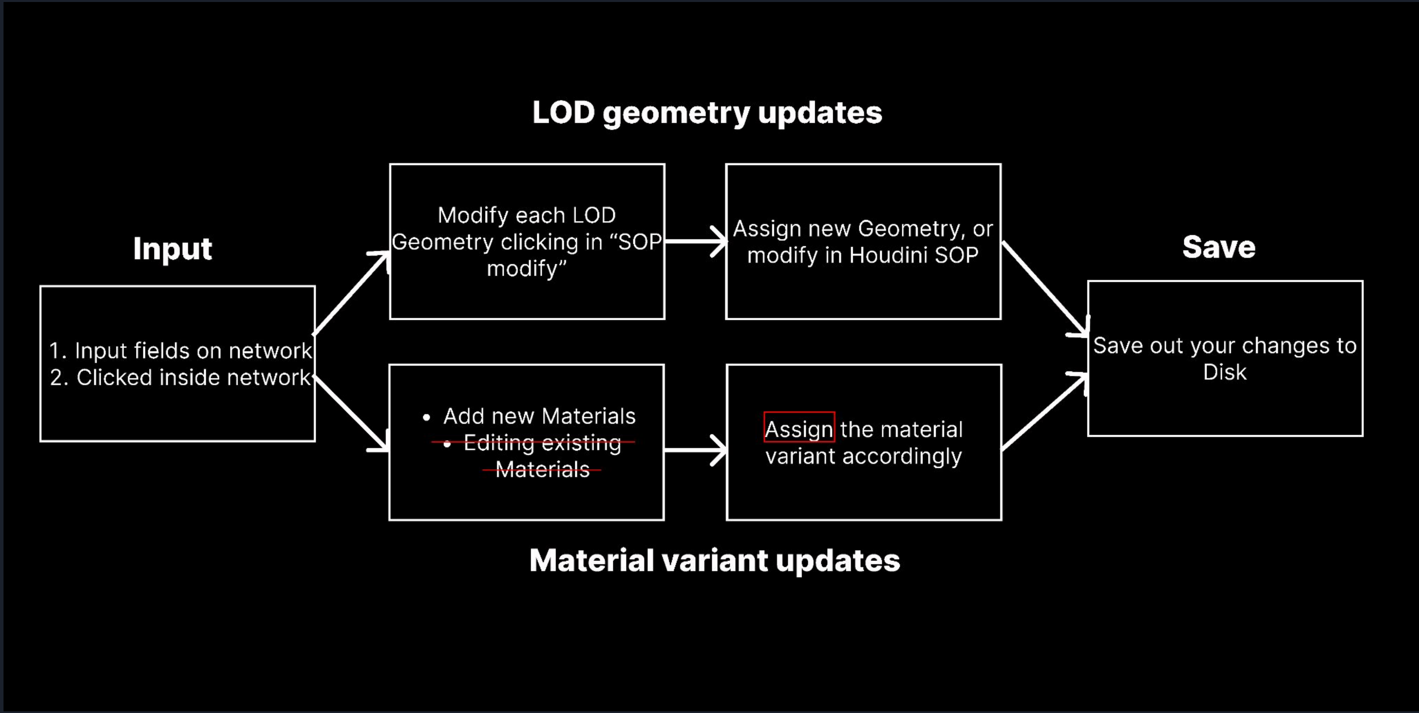
Save

Save out your changes to Disk

Update old asset

Before

After



Input

1. Input fields on network
2. Clicked inside network

LOD geometry updates

Modify each LOD Geometry clicking in “SOP modify”

Assign new Geometry, or modify in Houdini SOP

Material variant updates

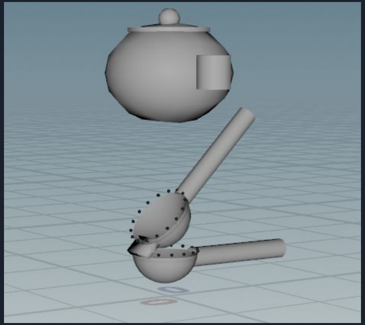
- Add new Materials
- ~~Editing existing Materials~~

Assign the material variant accordingly

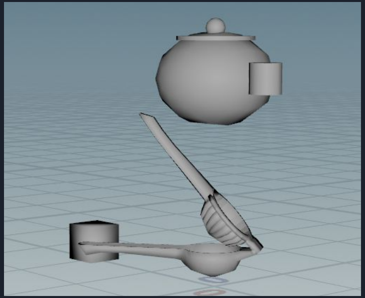
Save

Save out your changes to Disk

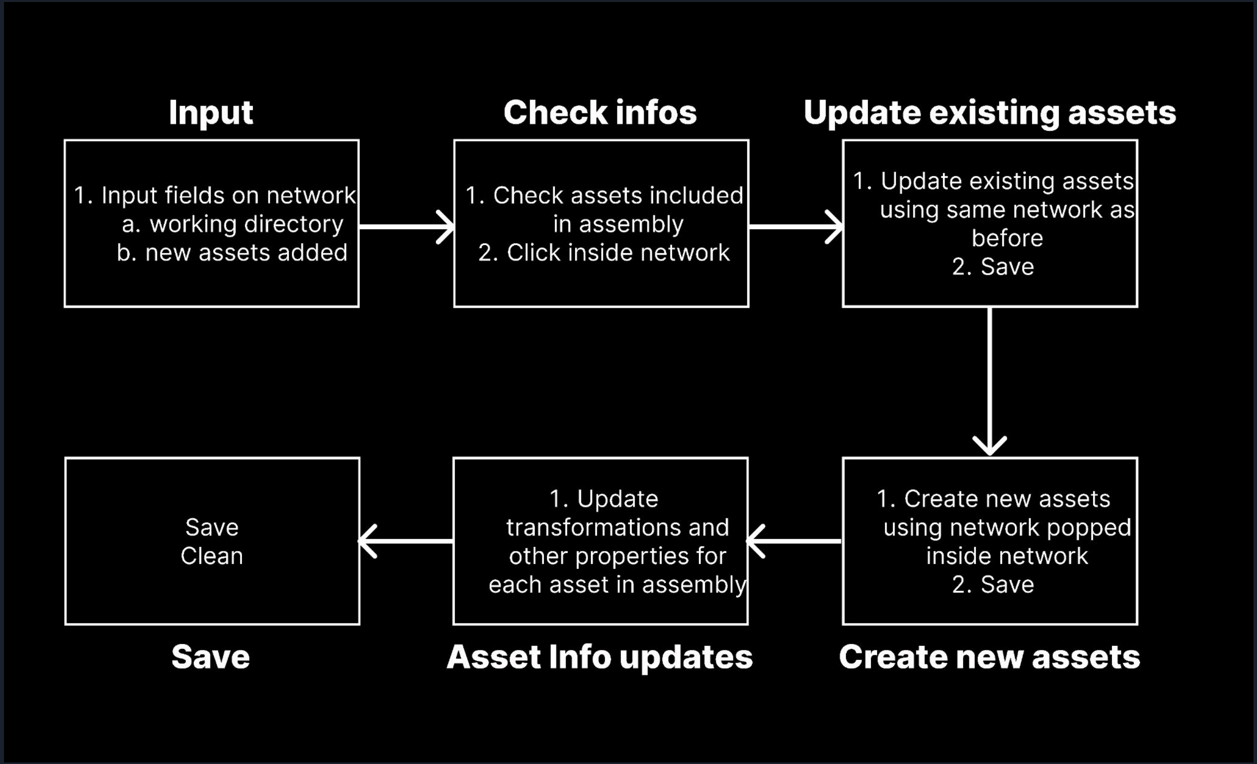
Update assembly



Before



After



Input

1. Input fields on network
   a. working directory
   b. new assets added

Check infos

1. Check assets included in assembly
2. Click inside network

Update existing assets

1. Update existing assets using same network as before
2. Save

Create new assets

1. Create new assets using network popped inside network
2. Save

Asset Info updates

1. Update transformations and other properties for each asset in assembly

Save

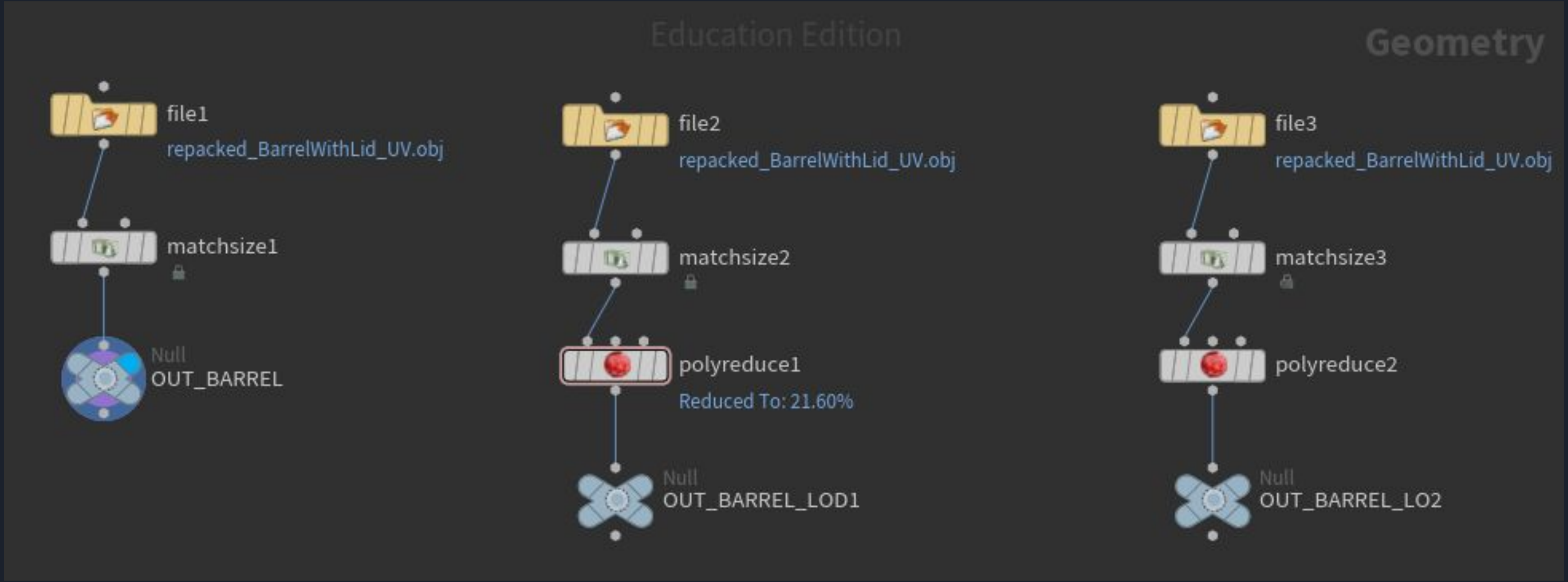
Save
Clean

Demo

LODs and Materials

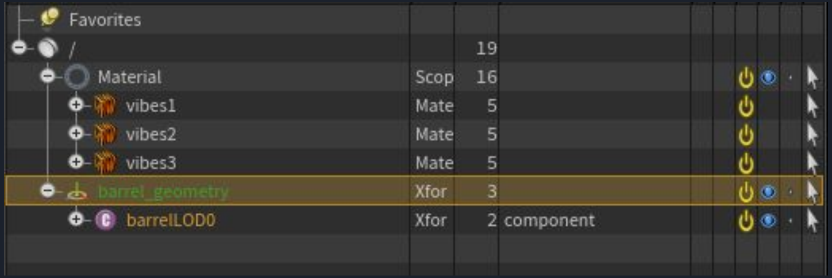
LODs

- Used Polyreduce nodes to create LOD variants

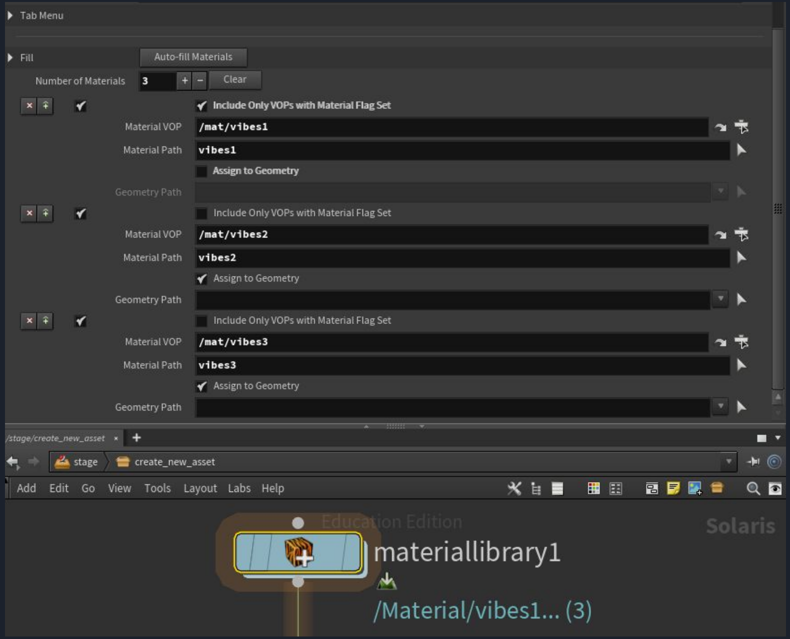


Materials

- Added material variants with material library node
- Materials are put into a variant such that they can be selected in the setvariant node



Favorites			
/		19	
Material	Scop	16	
vibes1	Mate	5	
vibes2	Mate	5	
vibes3	Mate	5	
barrel_geometry	Xfor	3	
barrelLOD0	Xfor	2	component



Tab Menu

Fill — Auto-fill Materials

Number of Materials: 3 — Clear

☑ Include Only VOPs with Material Flag Set
Material VOP: /mat/vibes1
Material Path: vibes1
□ Assign to Geometry
Geometry Path:

□ Include Only VOPs with Material Flag Set
Material VOP: /mat/vibes2
Material Path: vibes2
☑ Assign to Geometry
Geometry Path:

□ Include Only VOPs with Material Flag Set
Material VOP: /mat/vibes3
Material Path: vibes3
☑ Assign to Geometry
Geometry Path:

LODs and Materials .usda Files

Lighting

Lighting & Rendering Pipeline

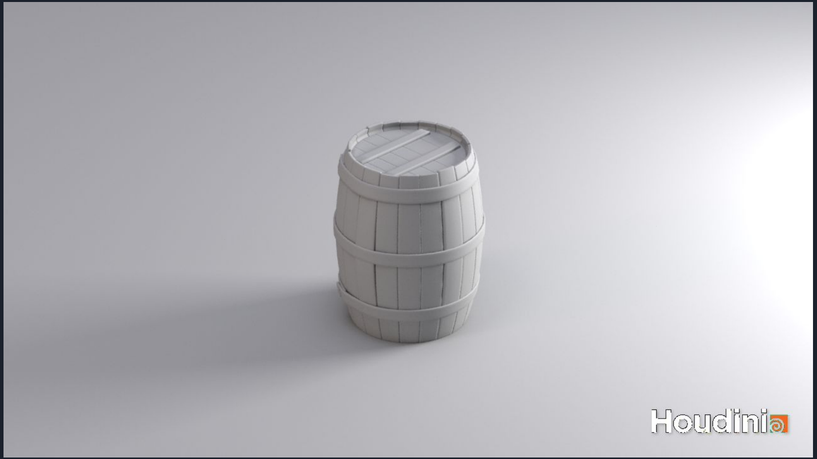
1. Hip file opening w/ metadata
2. USDA asset import
3. Neutral Lighting in Solaris
4. Rendering in Karma
5. Outputting to disk

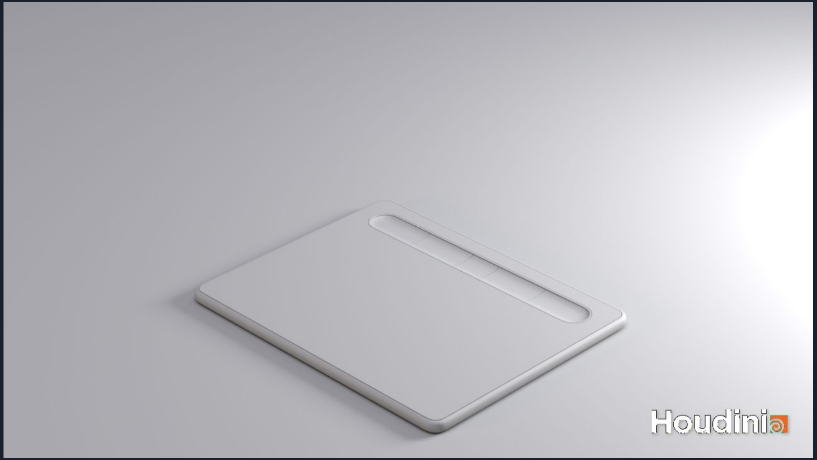
Blockers

- Griddle build issues
- xForm is not boundable

Future Steps

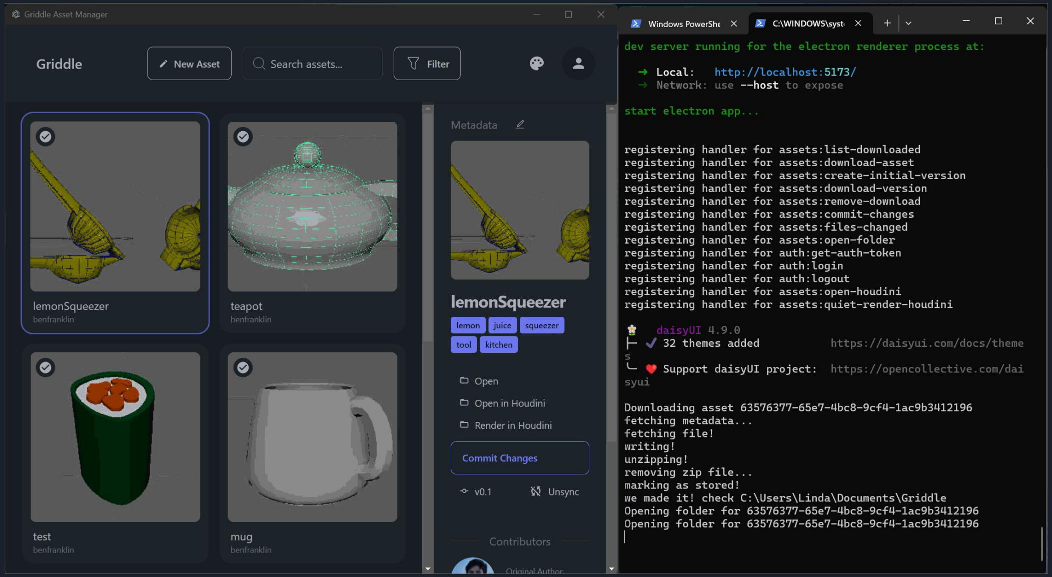
- Reinforcing asset scale & pivot standards
- Material instancing
- Dynamic camera adjustment to asset & meter scale measure in render
- Using the render image





Workflow Overview

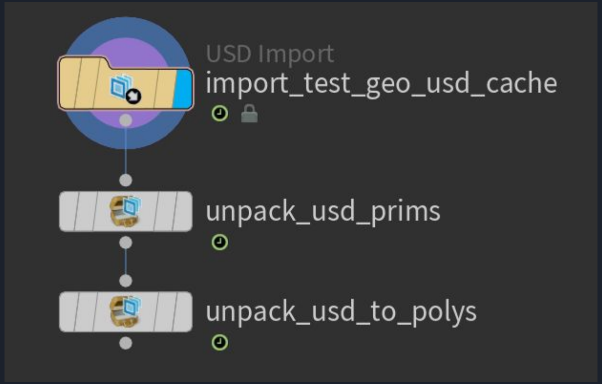
1. User click on “Render in Houdini” button
2. Houdini is launched headlessly
   a. Asset directory path mapped to file import & render output
   b. Karma rendering is started
3. Render image will be generated in <20 seconds from the XPU render



```
dev server running for the electron renderer process at:

  ➜  Local:   http://localhost:5173/
  ➜  Network: use --host to expose

start electron app...


registering handler for assets:list-downloaded
registering handler for assets:download-asset
registering handler for assets:create-initial-version
registering handler for assets:download-version
registering handler for assets:remove-download
registering handler for assets:commit-changes
registering handler for assets:files-changed
registering handler for assets:open-folder
registering handler for auth:get-auth-token
registering handler for auth:login
registering handler for auth:logout
registering handler for assets:open-houdini
registering handler for assets:quiet-render-houdini

🌼   daisyUI 4.9.0
├─ ✔︎ 32 themes added          https://daisyui.com/docs/theme
s
╰─ ❤︎ Support daisyUI project:  https://opencollective.com/dai
syui

Downloading asset 63576377-65e7-4bc8-9cf4-1ac9b3412196
fetching metadata...
fetching file!
writing!
unzipping!
removing zip file...
marking as stored!
we made it! check C:\Users\Linda\Documents\Griddle
Opening folder for 63576377-65e7-4bc8-9cf4-1ac9b3412196
Opening folder for 63576377-65e7-4bc8-9cf4-1ac9b3412196
```

USD Import and Conversions to Polygons

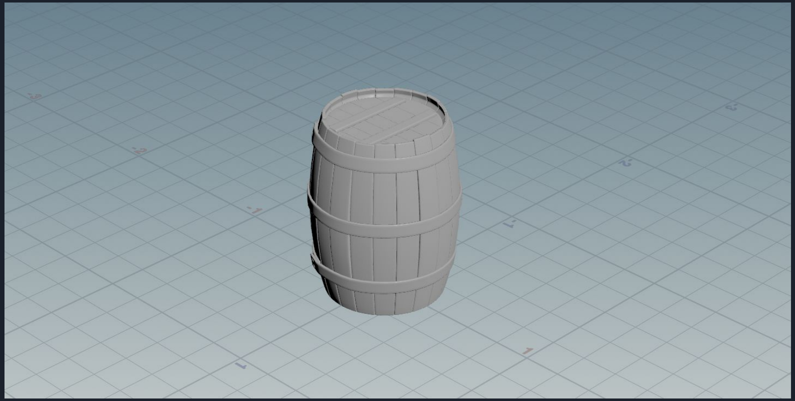
- Automatic USDA asset import based on asset directory name
- Convert to polygon for compatibility & performance



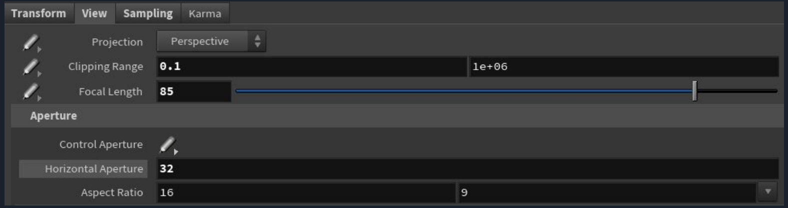
USD Import
import_test_geo_usd_cache

unpack_usd_prims

unpack_usd_to_polys



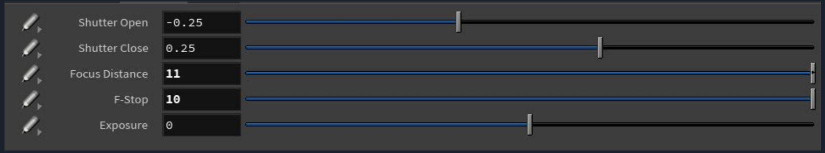
Camera + Neutral 3-Point Lighting w/ Skybox in Solaris



Projection: Perspective
Clipping Range: 0.1 1e+06
Focal Length: 85

Aperture

Control Aperture
Horizontal Aperture: 32
Aspect Ratio: 16 9



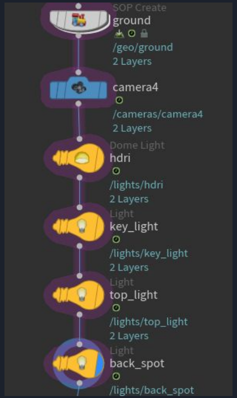
Shutter Open: -0.25
Shutter Close: 0.25
Focus Distance: 11
F-Stop: 10
Exposure: 0

Camera

- Telephoto perspective
  - Still the “isometric look, but more natural”
- Big F-Stop for no Depth of Field

Lighting

- HDRI
- Key light
- Top light
- Bounce



SOP Create
ground
/geo/ground
2 Layers

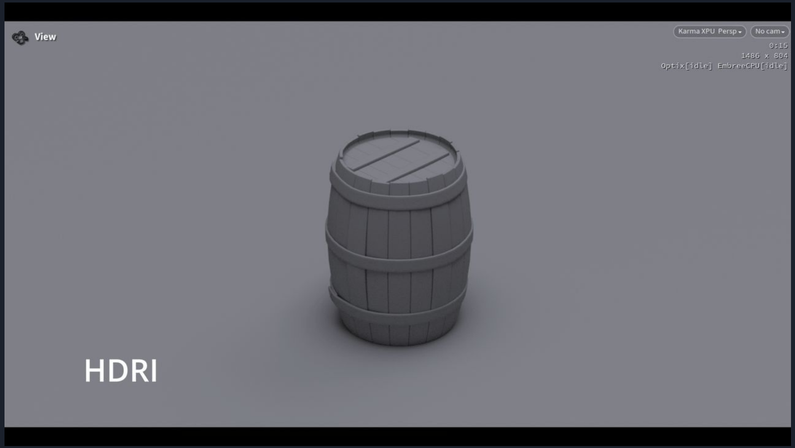
camera4
/cameras/camera4
2 Layers

Dome Light
hdri
/lights/hdri
2 Layers

Light
key_light
/lights/key_light
2 Layers

Light
top_light
/lights/top_light
2 Layers

Light
back_spot
/lights/back_spot



Rendering with Karma XPU

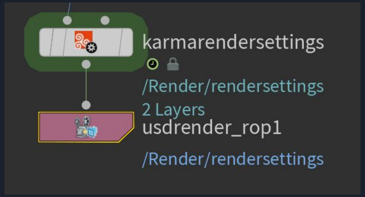
Resolution

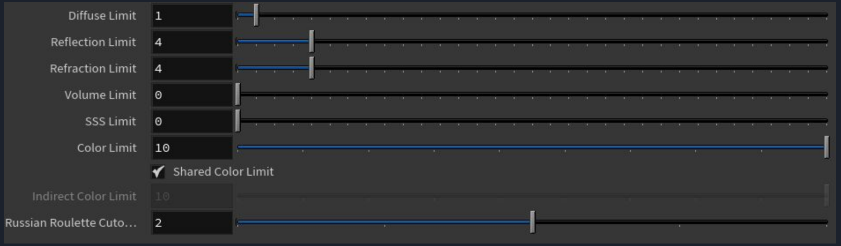
- 1280 by 720 (16:9)

Render Engine

- XPU (CPU + GPU)
  - 64 samples with the following setting:

Saving to Disk

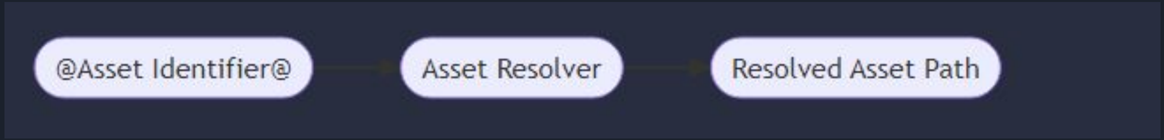




Diffuse Limit 1
Reflection Limit 4
Refraction Limit 4
Volume Limit 0
SSS Limit 0
Color Limit 10
☑ Shared Color Limit
Indirect Color Limit 10
Russian Roulette Cuto... 2

Asset Resolver

What is an Asset Resolver?

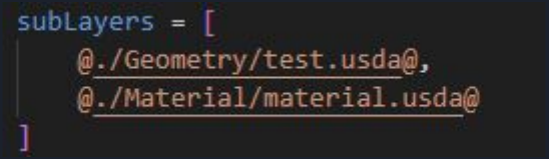


@Asset Identifier@

Asset Resolver

Resolved Asset Path

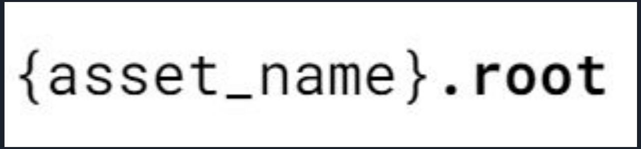
- In USD, anything with the `@` around it
- By default, pxr uses file paths on disk to resolve assets
- We can define our own by creating an asset resolver plugin
- https://openusd.org/release/api/ar_page_front.html
- `ArResolver` and `ArAsset`
  - pxr/usd/ar/resolver.h
  - pxr/usd/ar/asset.h



```
subLayers = [
    @./Geometry/test.usda@,
    @./Material/material.usda@
]
```

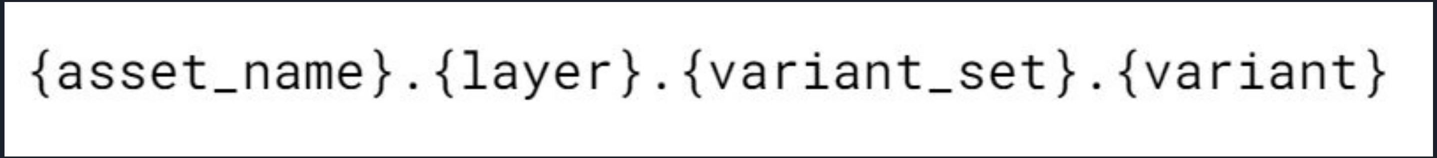
Our Asset Resolver

Entry point to asset:



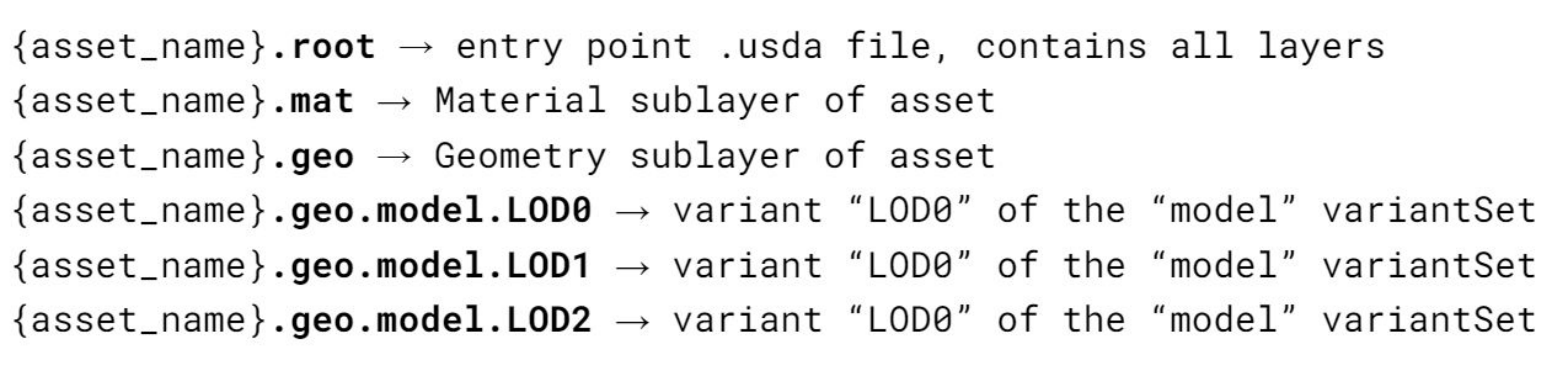
```
{asset_name}.root
```

Specify asset layer/variant



```
{asset_name}.{layer}.{variant_set}.{variant}
```

Our Asset Resolver



{asset_name}**.root** → entry point .usda file, contains all layers

{asset_name}**.mat** → Material sublayer of asset

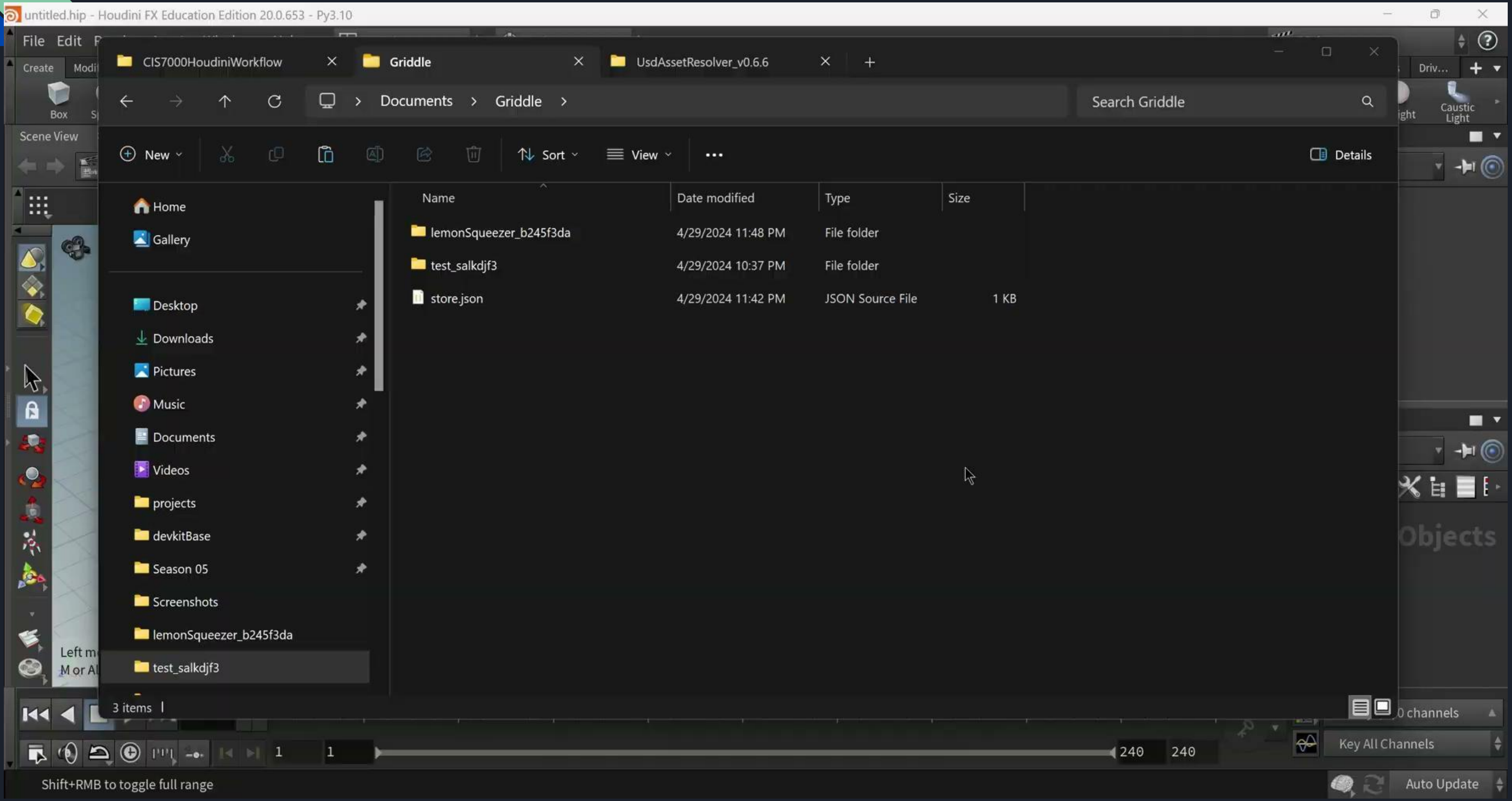
{asset_name}**.geo** → Geometry sublayer of asset

{asset_name}**.geo.model.LOD0** → variant “LOD0” of the “model” variantSet

{asset_name}**.geo.model.LOD1** → variant “LOD0” of the “model” variantSet

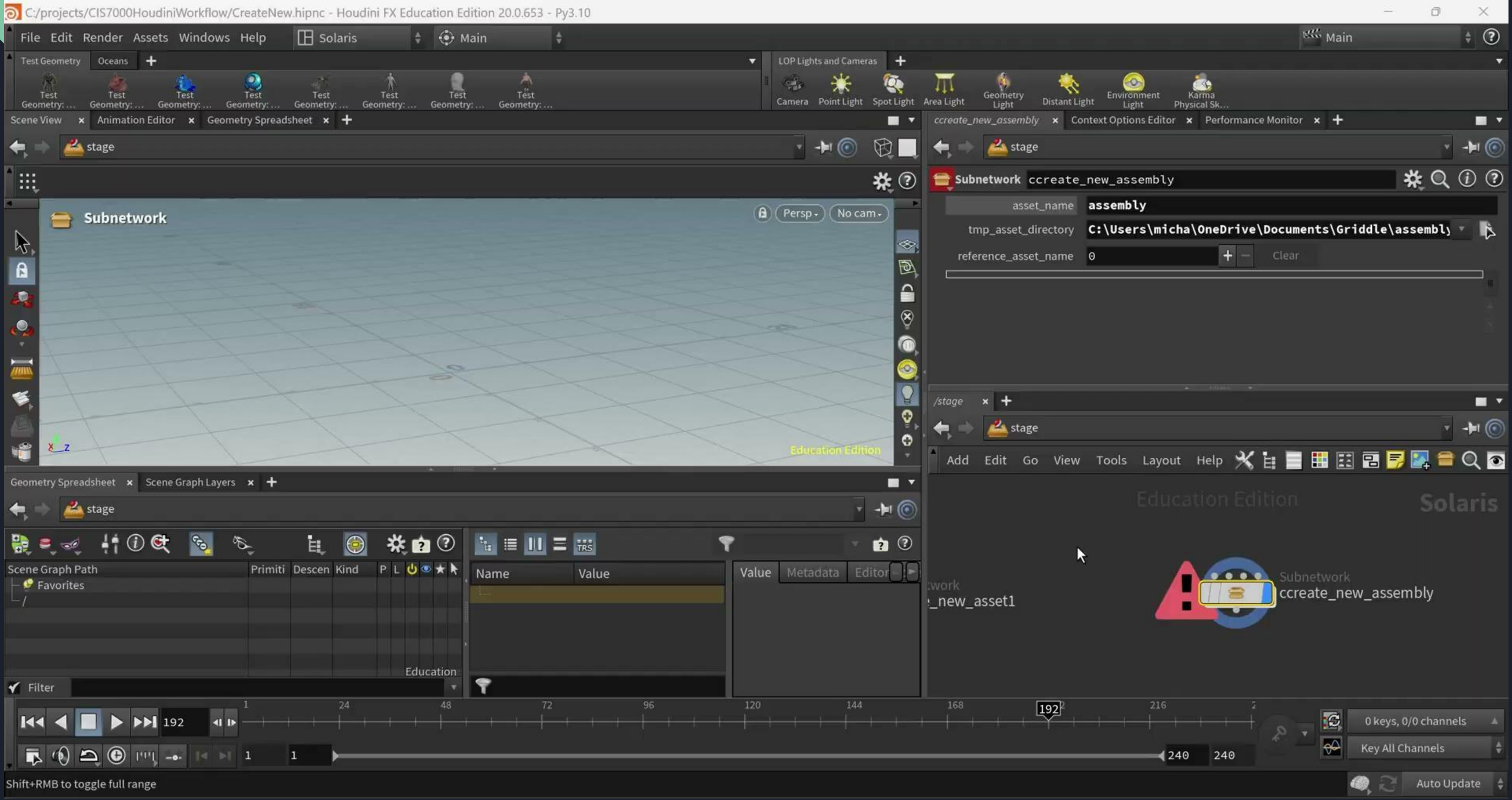
{asset_name}**.geo.model.LOD2** → variant “LOD0” of the “model” variantSet

Demo



Name	Date modified	Type	Size
lemonSqueezer_b245f3da	4/29/2024 11:48 PM	File folder	
test_salkdjf3	4/29/2024 10:37 PM	File folder	
store.json	4/29/2024 11:42 PM	JSON Source File	1 KB

Demo (Assemblies)



C:/projects/CIS7000HoudiniWorkflow/CreateNew.hipnc - Houdini FX Education Edition 20.0.653 - Py3.10

Subnetwork ccreate_new_assembly

asset_name: assembly

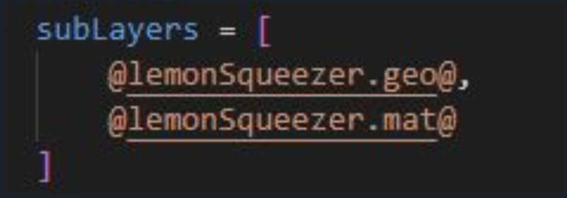
tmp_asset_directory: C:\Users\micha\OneDrive\Documents\Griddle\assembly

reference_asset_name: 0

Shift+RMB to toggle full range

Asset Resolver Further Work

- Export USD’s with custom asset identifier
- Version pinning (latest? absolute versions?)
- Use more robust API (define an ArAsset + ArResolver)
  - fetch directly from our database
- Maya integration



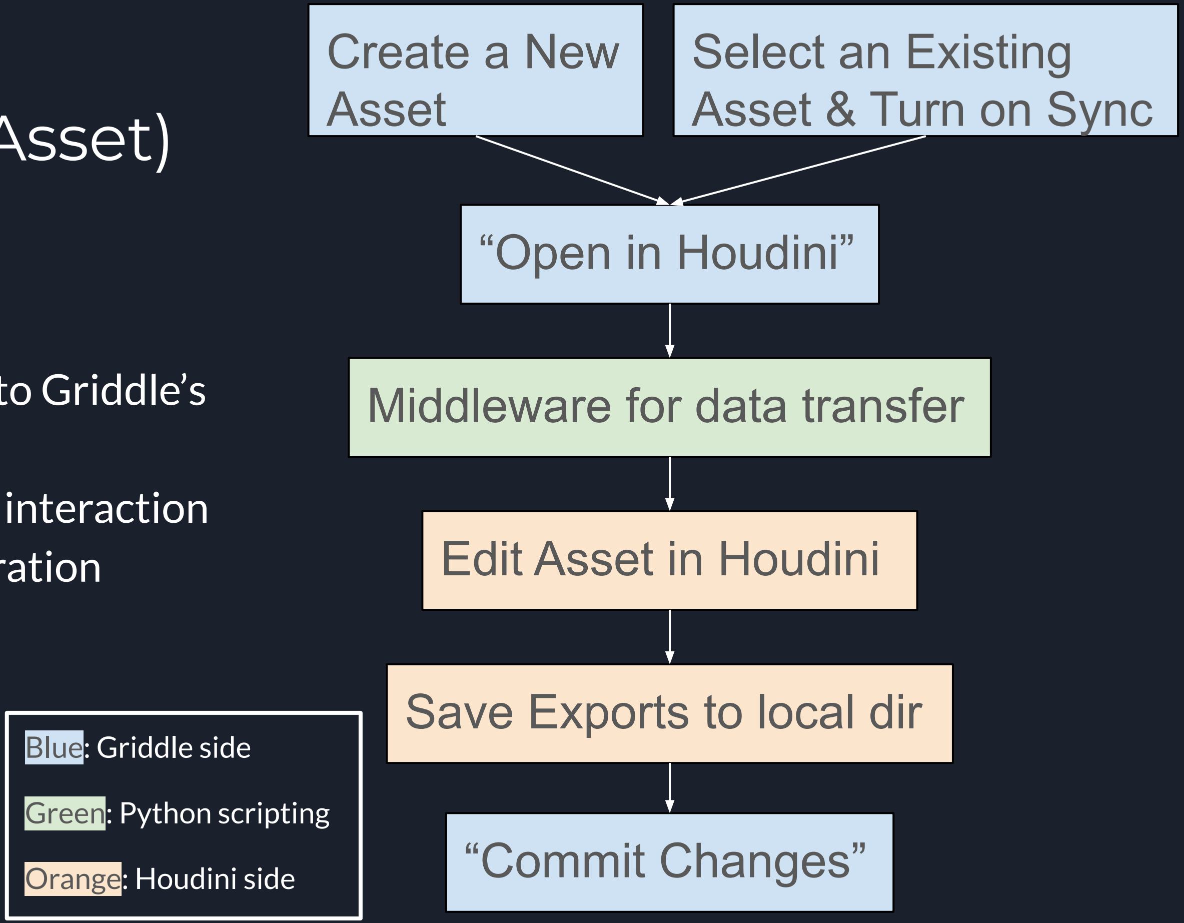
```
subLayers = [
    @lemonSqueezer.geo@,
    @lemonSqueezer.mat@
]
```

Griddle Integration

Overview (Component Asset)

Advantages:

- Maximally adhere to Griddle’s workflow
- Separate database interaction and 3D asset generation



Create a New Asset

Select an Existing Asset & Turn on Sync

“Open in Houdini”

Middleware for data transfer

Edit Asset in Houdini

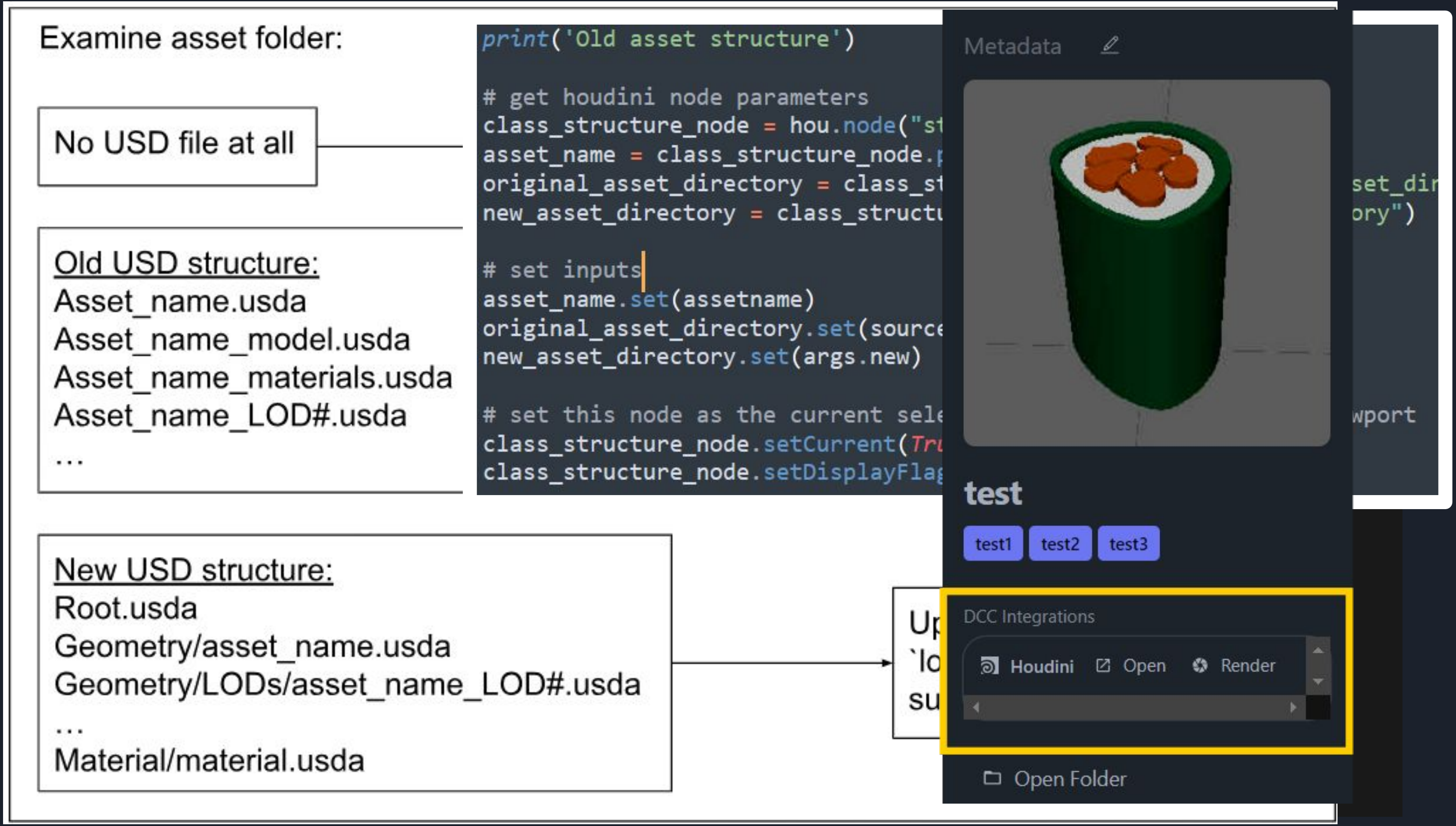
Save Exports to local dir

“Commit Changes”

Blue: Griddle side

Green: Python scripting

Orange: Houdini side



Examine asset folder:

No USD file at all

Old USD structure:
Asset_name.usda
Asset_name_model.usda
Asset_name_materials.usda
Asset_name_LOD#.usda
…

New USD structure:
Root.usda
Geometry/asset_name.usda
Geometry/LODs/asset_name_LOD#.usda
…
Material/material.usda

Up
`lo
su

```
print('Old asset structure')

# get houdini node parameters
class_structure_node = hou.node("st
asset_name = class_structure_node.p
original_asset_directory = class_st
new_asset_directory = class_structu

# set inputs
asset_name.set(assetname)
original_asset_directory.set(source
new_asset_directory.set(args.new)

# set this node as the current sele
class_structure_node.setCurrent(Tru
class_structure_node.setDisplayFlag
```

```
set_dir
ory")

wport
```

Houdini Command-line Scripting & Hython

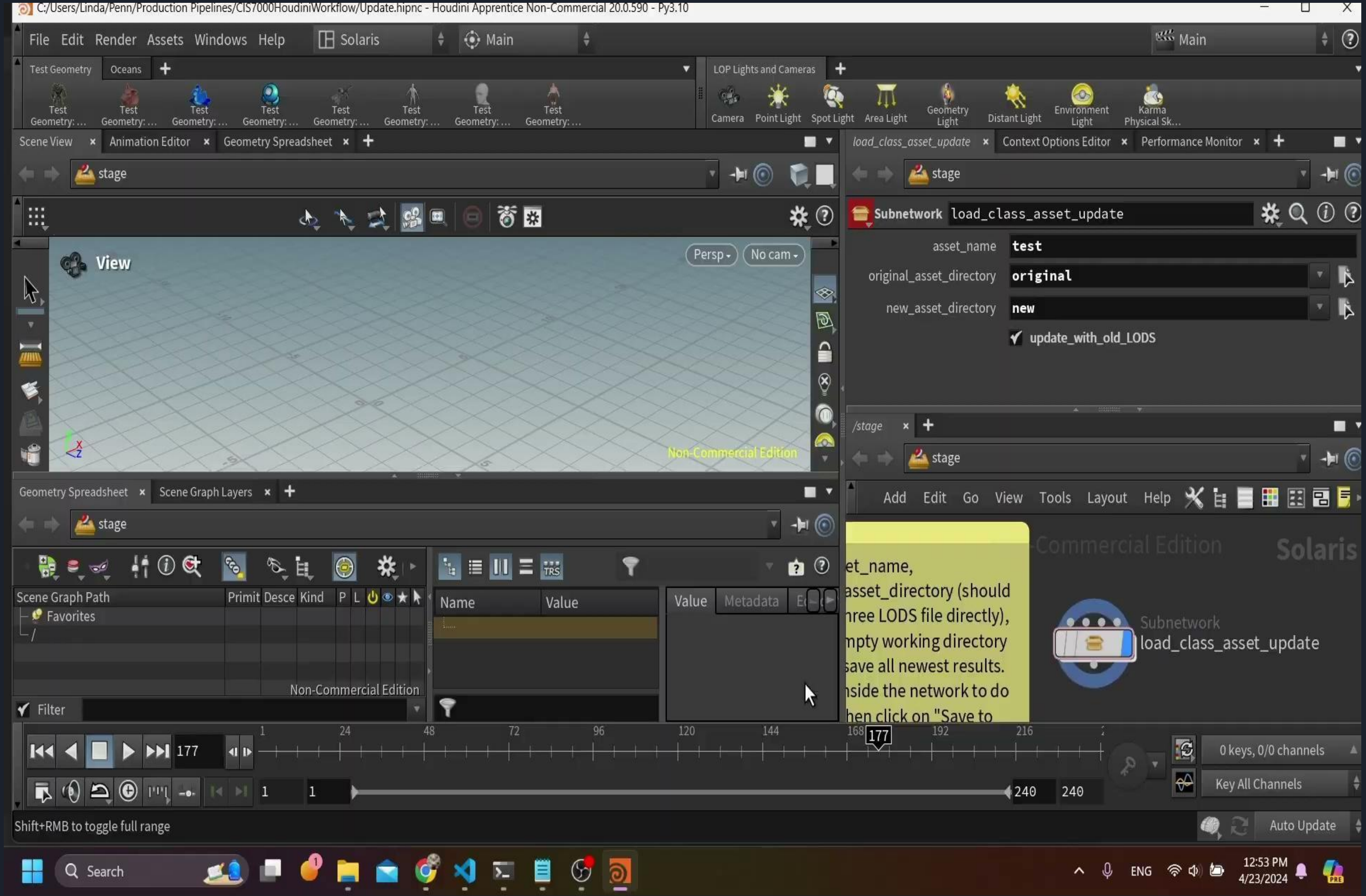
- `$houdini template.hip launchTemplate.py [optional flags to python]`

  This way of launching Houdini is the special command line tool that **REQUIREs** the user to set `$HFS` system environment variable to their local houdini installation path so we can locate the specific Houdini version and executable to run.

- `$hython script.py [optional flags to python]`

  Hython is a Python shell that ships with Houdini that is slightly different from the standard Python shell. It can launch Houdini headlessly without UI so it's perfect for the lighting template, which contains an automation for universal lighting, to render out an image.

First iteration:



C:/Users/Linda/Penn/Production Pipelines/CIS7000HoudiniWorkflow/Update.hipnc - Houdini Apprentice Non-Commercial 20.0.590 - Py3.10

Subnetwork load_class_asset_update

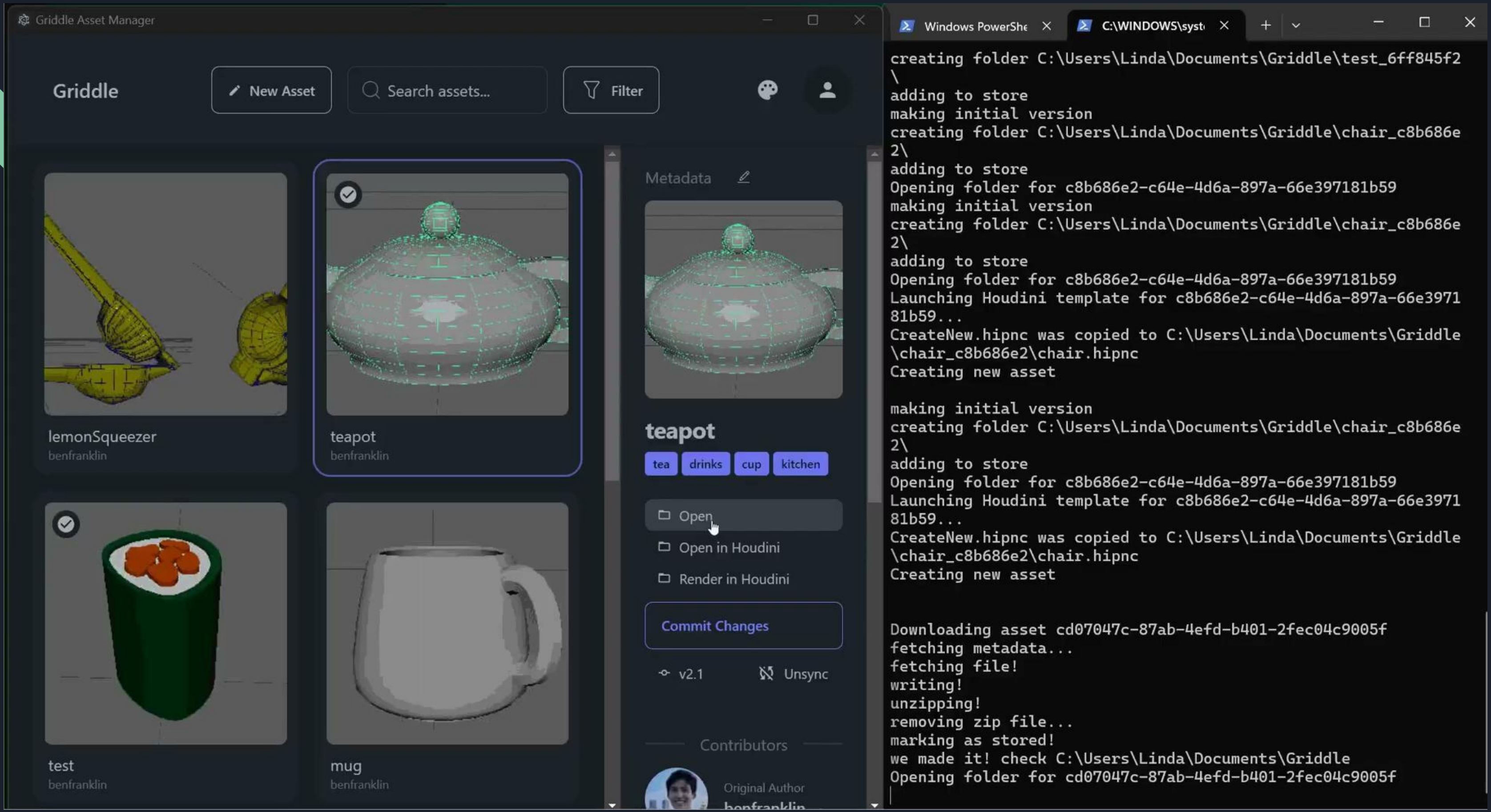
asset_name test

original_asset_directory original

new_asset_directory new

✓ update_with_old_LODS

et_name,
asset_directory (should
hree LODS file directly),
mpty working directory
save all newest results.
nside the network to do
hen click on "Save to



```
creating folder C:\Users\Linda\Documents\Griddle\test_6ff845f2
\
adding to store
making initial version
creating folder C:\Users\Linda\Documents\Griddle\chair_c8b686e
2\
adding to store
Opening folder for c8b686e2-c64e-4d6a-897a-66e397181b59
making initial version
creating folder C:\Users\Linda\Documents\Griddle\chair_c8b686e
2\
adding to store
Opening folder for c8b686e2-c64e-4d6a-897a-66e397181b59
Launching Houdini template for c8b686e2-c64e-4d6a-897a-66e3971
81b59...
CreateNew.hipnc was copied to C:\Users\Linda\Documents\Griddle
\chair_c8b686e2\chair.hipnc
Creating new asset

making initial version
creating folder C:\Users\Linda\Documents\Griddle\chair_c8b686e
2\
adding to store
Opening folder for c8b686e2-c64e-4d6a-897a-66e397181b59
Launching Houdini template for c8b686e2-c64e-4d6a-897a-66e3971
81b59...
CreateNew.hipnc was copied to C:\Users\Linda\Documents\Griddle
\chair_c8b686e2\chair.hipnc
Creating new asset


Downloading asset cd07047c-87ab-4efd-b401-2fec04c9005f
fetching metadata...
fetching file!
writing!
unzipping!
removing zip file...
marking as stored!
we made it! check C:\Users\Linda\Documents\Griddle
Opening folder for cd07047c-87ab-4efd-b401-2fec04c9005f
```

Thank you

Houdini is awesome. Solaris is fantastic. USD…?