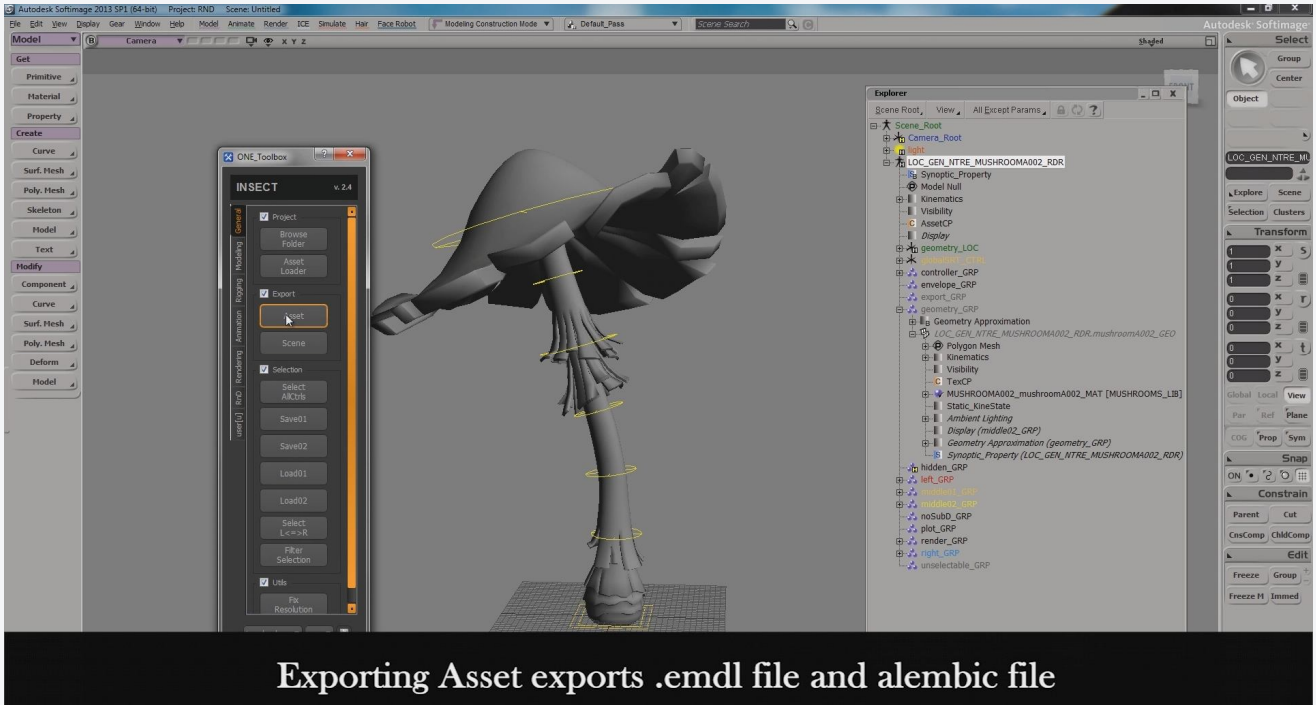
Show Reel Breakdown - Project Insectibles (Animated TV Series Pilot Episode)

00:04 - 00:10

- Built the Softimage to Houdini Asset Pipeline using Houdini Digital Assets
- Built the Mudbox PTex Texturing Pipeline using the Mudbox SDK.
- Contributed to Lighting few shots, Shading Characters, Environments and Props and Rendering.

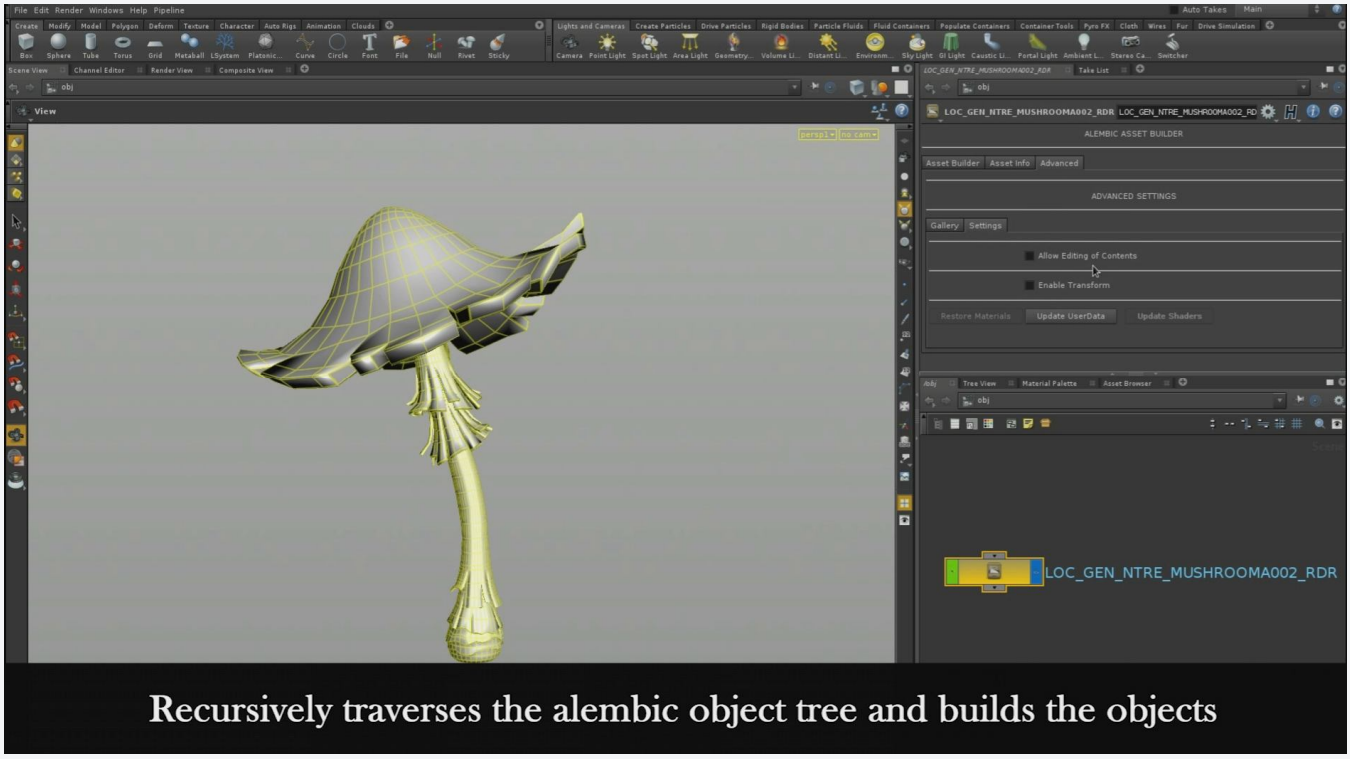
00:11 - 00:18

- Translation of Softimage Asset (Mushroom) into Production ready Houdini Asset for Rendering.
- Export Softimage Asset exports a Model (.emdl) file and Alembic (.abc) file along with an XML file containing information about the Material and Texture assignments



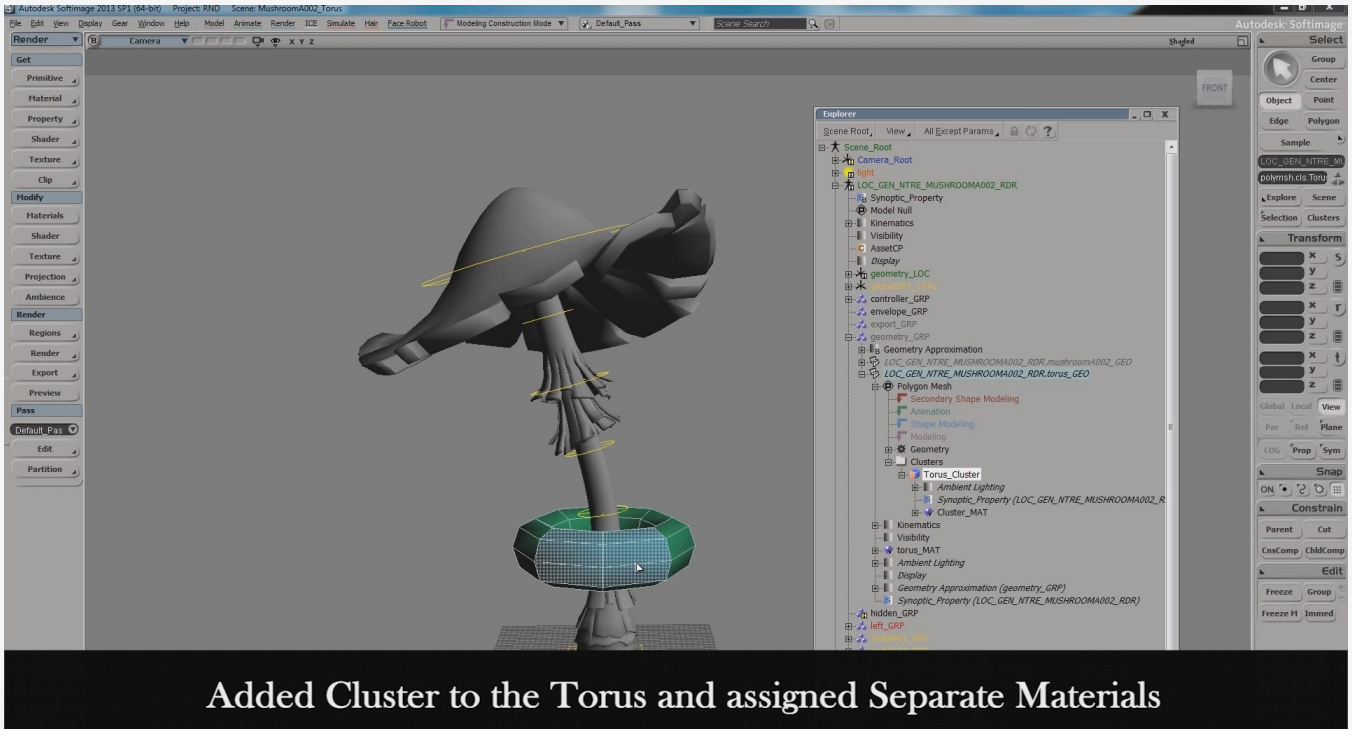
00:19 - 00:34

- Creation of the Exported Mushroom Asset in Houdini.
- Creates the Alembic Xform Nodes with Custom Geometry node with Materials assigned properly
- For each material created in Softimage, it creates a corresponding Houdini material and assigns the Textures using the information from the XML.
- Exporting the Houdini Asset exports an OTL file.



00:34 - 00:40

- In Softimage, added a torus object with clusters and separate materials for clusters.



00:41 - 00:55

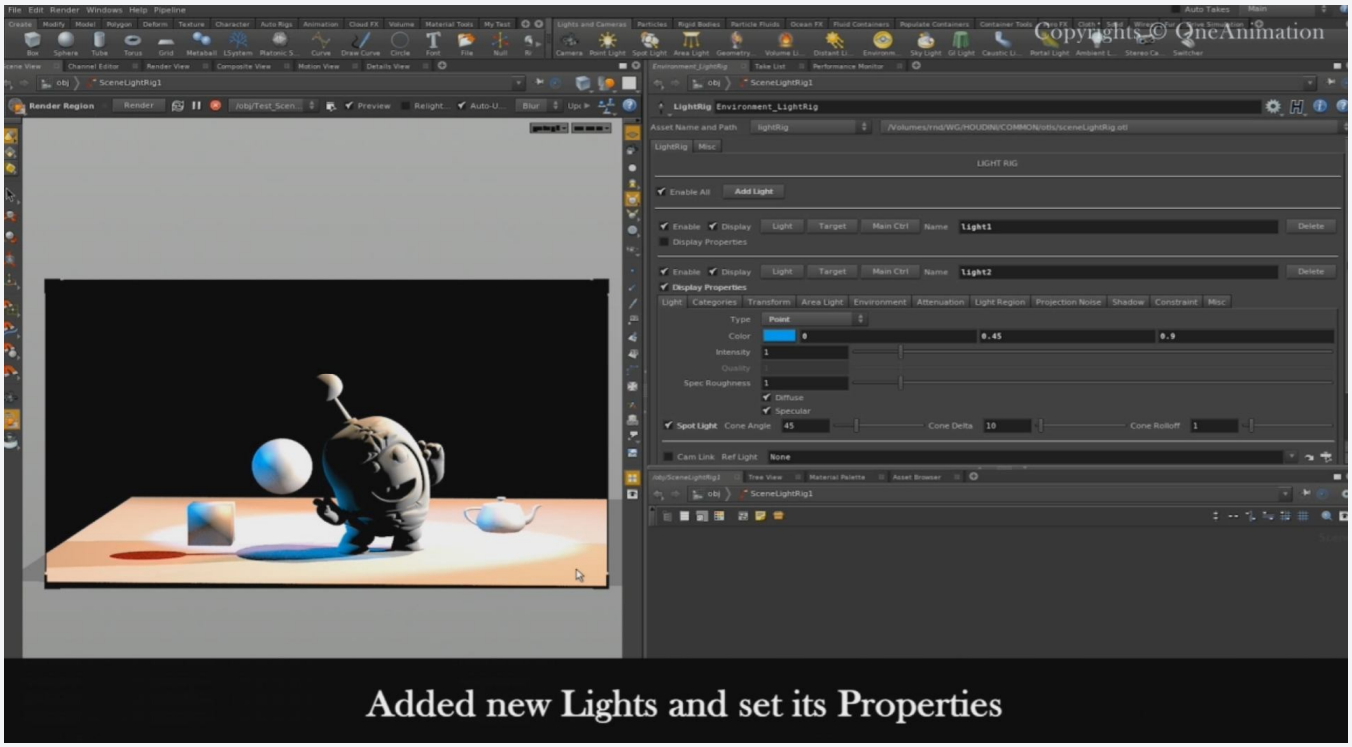
- Deleted the Houdini Asset we exported before and Re-Importing the Asset again.
- On Import, it checks for newly added objects and materials and creates them and assigns them properly. In this way, Houdini Asset is always synchronized with the Softimage Asset.
- Since we added clusters to the torus, it creates a separate material node inside the Geometry node and sets the cluster as group name and cluster material as its material.
- Update Hierarchy Button, an option to update the objects and materials to the current Softimage asset.

00:56 - 01:09

- TurnTable LightRig for Assets to give the assets a default lighting setup while shading.
- Final Render of Insectibles.

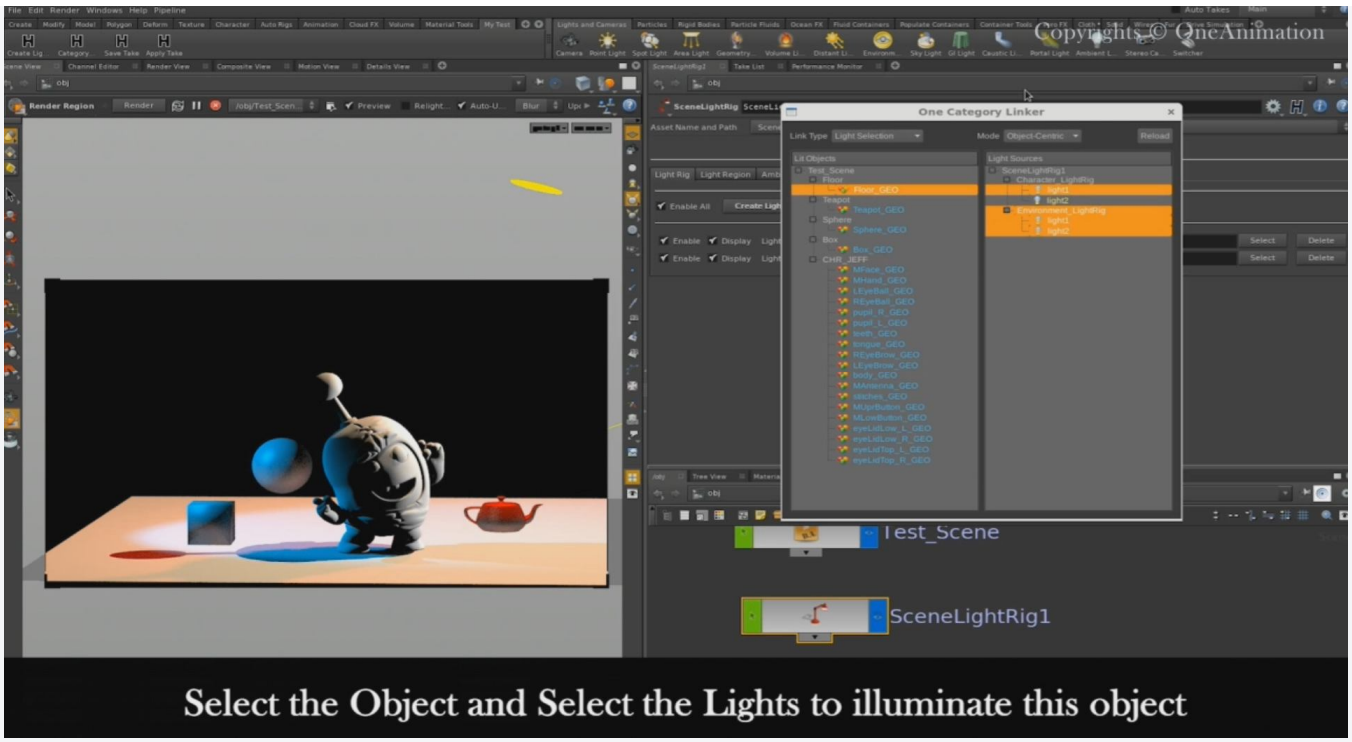
01:10 - 01:29

- Custom LightRig for Scene and Light Linking tools demonstrated using a Test Scene.
- Creation of the main LightRig creates a Lightrig with a light with default lighting setup.
- LightRig has all the properties of all the lights created.
- Adjusted the transforms and other properties of the Light.
- Ability to Create/delete as many Lights and Lightrigs.



01:30 - 01:58

- Light Linker tool built in PyQt. It works similar to the Houdini's built in Light Linker but links lights and objects based on Categories.
- Select the Light from the Left tree and select the objects from the Right tree to be illuminated by the selected light.
- Once the Light and object is linked, then the selection of light shows all the objects illuminated by this light.
- Similarly switch to Shadow selection and select the Light in the Left tree and select the Object in the Right Tree to cast shadows from this light.
- Has Object centric and Light centric modes similar to Houdini Light Linker for easy linking of objects with Lights
- Custom Geometry Node we used for all assets has some special parameters with expressions that evaluates to the Category names we used in Lights.



01:58 - 02:25

- Houdini Shader Library Tool built in PyQt
- Selected the Material node to add to the Library. Render the image for previewing.
- Set the Material name, Category for the Gallery entry and comments and add it to the Library.
- Applied the parameters from the Shader in the Library to the newly created material.