FTV463337

Material Transfer in Production using Arnold and MaterialX

Krishnan Ramachandran
Autodesk

Learning Objectives

- Transport your looks across supported software easily.
- Build Modify and Edit your looks all along your pipeline.
- Scale your pipeline combining modular looks across software.

Description

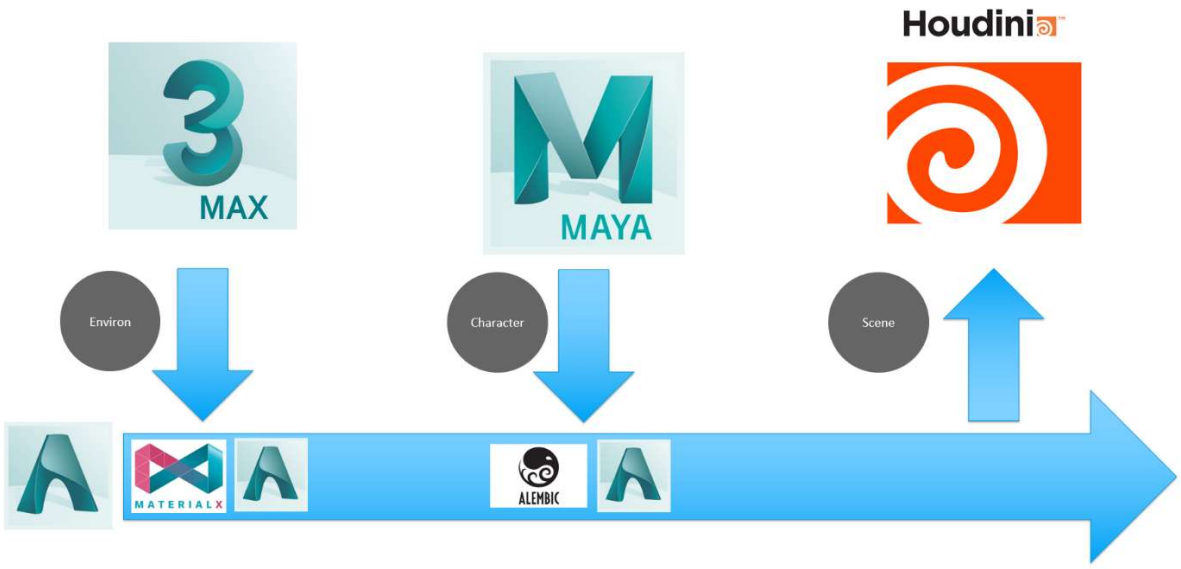
Arnold solves the problem of being able transfer asset materials across different software products along various stages of a pipeline without any loss of fidelity. Arnold users are free to use a software tool of their choice without worrying about their asset looks.

Speaker

Krishnan is a Senior Software Engineer with the Arnold Rendering team at Autodesk . His focus is plugin development for Arnold. He comes with nearly 10 years' experience working in the animation and VFX industries.He has held different roles within the VFX pipeline working in departments like Character Simulation, Procedural generation and Rendering. He brings his experience in pipeline and tools development to help solve common problems in the VFX pipeline.Outside of work, Krishnan enjoys spending most of his time with his new found love for gardening, carpentry and coffee.

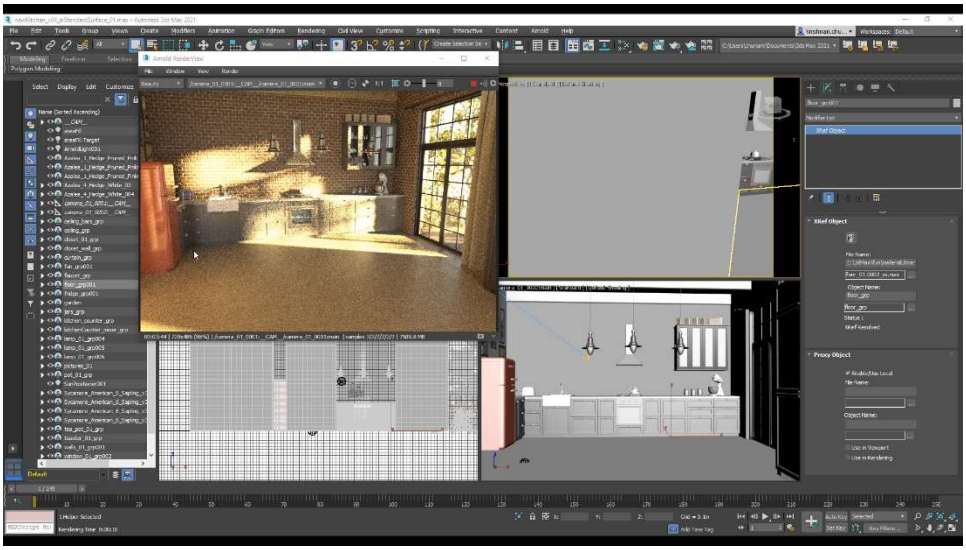
Arnold as the Data Driver for your pipeline

A modern VFX rendering pipeline is made up of many materials with detailed descriptions called looks. Looks need to be passed across different software products without loss of fidelity. Current workflows involve duplication of these looks to suit each software's requirements through complex pipeline tasks that are hard and laborious to maintain. Arnold lets you author asset looks in any supported software and then re-use the same looks in another software without any loss of fidelity and functionality using Arnold's own data format, which is fully extensible. Arnold furthers this approach by leveraging open-source back-ends like MaterialX and USD enabling users to build a very flexible, re-usable and connected pipeline. Arnold users can take advantage of this interoperability right out of the box. The data is always live, enabling users to make changes however drastic or minor at any point in the pipeline.

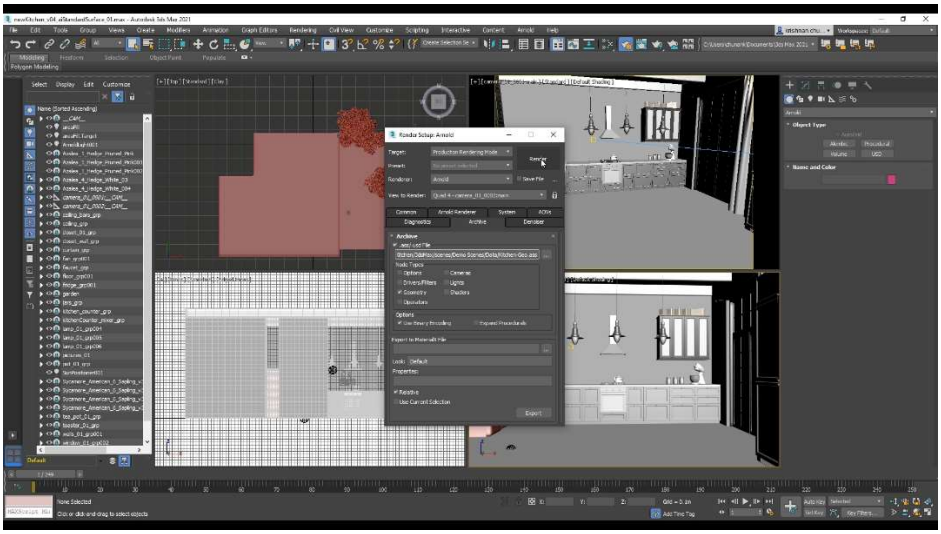


In this talk we will render a production level scene, with data that travels across 3 different software products. The environment will be created in 3DsMax , the character in Maya and the scene rendered out of Houdini. In each software the data will be exported to Arnold. We will look at how to bring them into each software without loss of any data while still retaining the ability to make changes to the underlying looks.

Arnold in Max: MaxToA

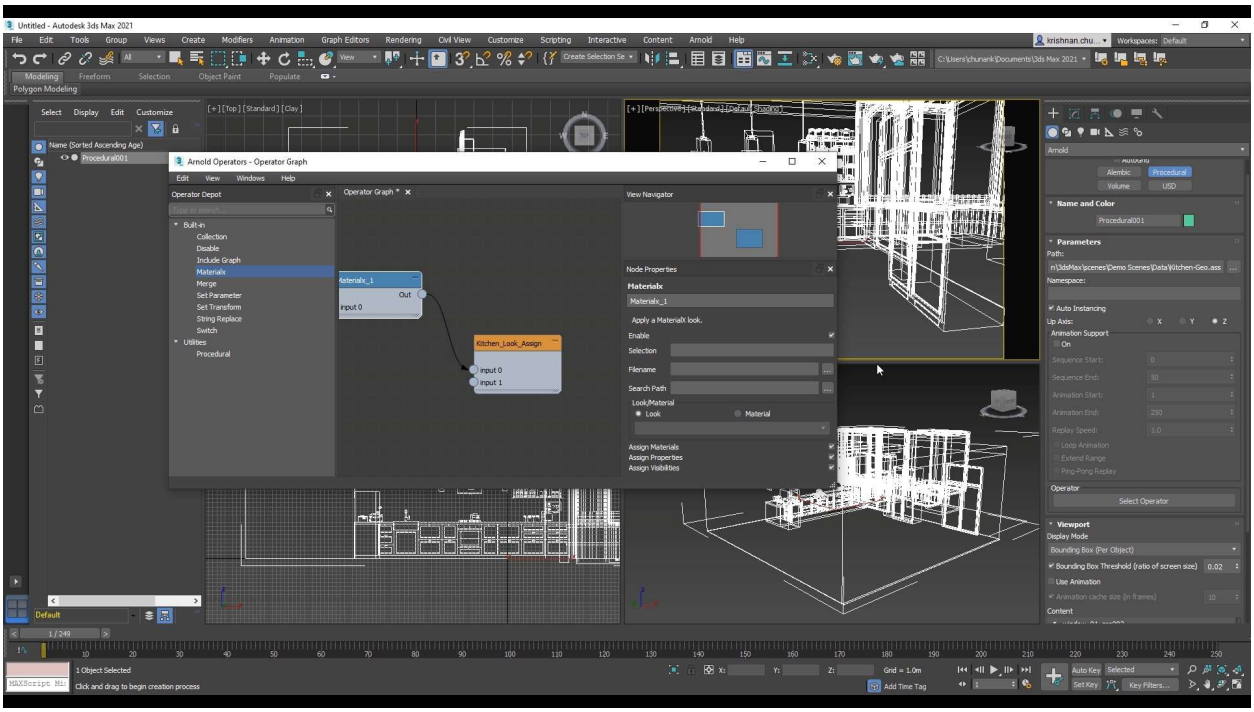


Arnold is fully interrated within Max with the MaxToA plugin. Arnold is also the default renderer in Max. With this intergration, we are able to take an environment that was modelled and look-dev'ed in Max and export it to Arnold.



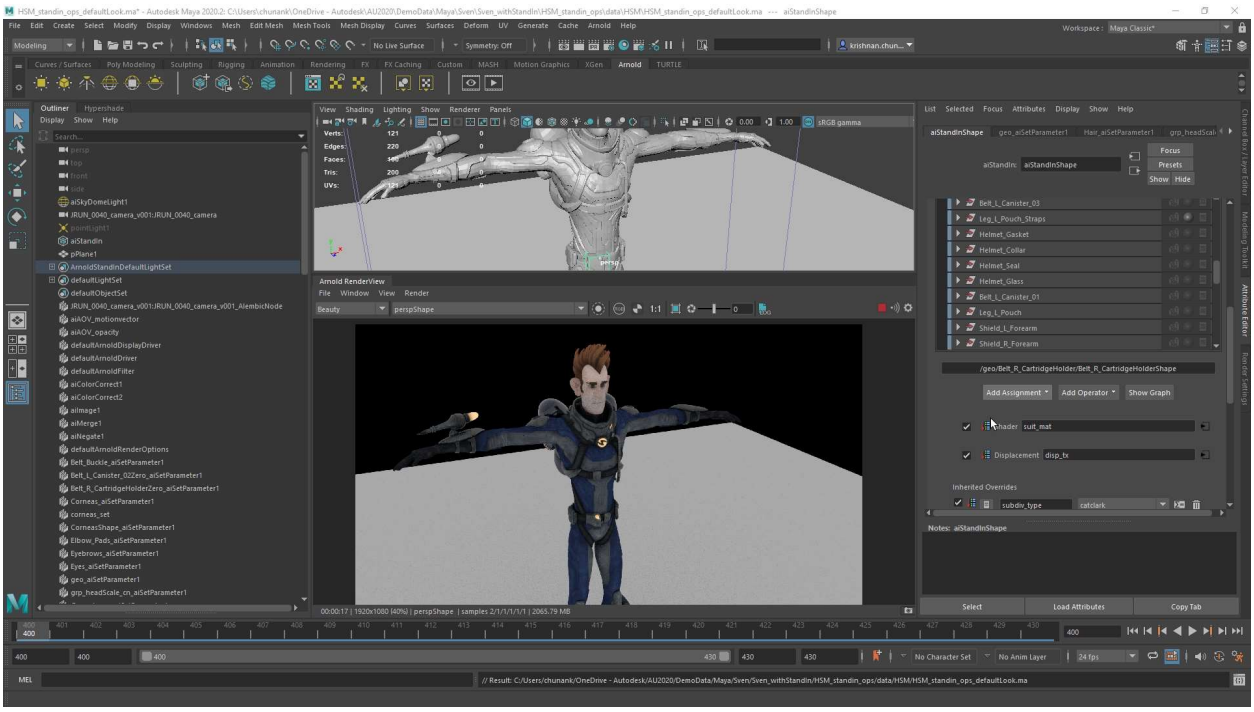
We export the geometry as an Arnold file with the .ass extension. This could very well be any geometry file format that Arnold supports like alembic or usd. We take all of the material assignments and export that as a MaterialX file. The options to do that can be found in the Render Setup menu.

We can bring these two pieces of geometry separaterly into the Max scene and assign the look MaterialX file to the geometry file. This way of de-coupling the geometry from the look can be extremely helpful. The geometry and look-development can continue to evolve independatntly of one another as long as the rules that have been set for the binding are not broken. The assignment happen via the use of Arnold Operators. Operators give you a way of being able control your scene at render time. Learn more about Operators in MaxToA here.

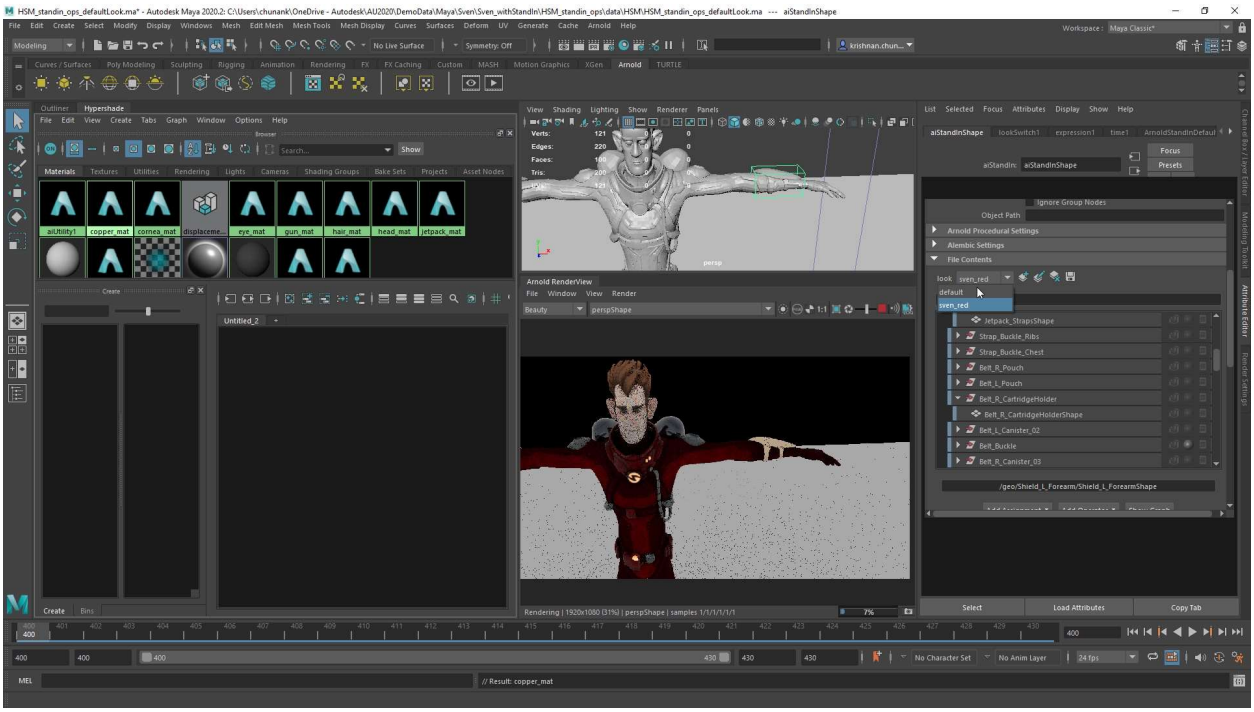


Arnold in Maya (MtoA)

Arnold has been the default renderer in Maya for quite a while now. In Maya, we explore a different way of assigning materials to geometry. We assign materials to geometry that's an alembic file. We bring the alembic into the scene as a StandIn (which is an MtoA specific node type) and use the StandIn viewer to inspect the contents of the alembic file and also assign materials to it. You can drag and drop shaders when you create a shader assignment using the StandIn viewer.



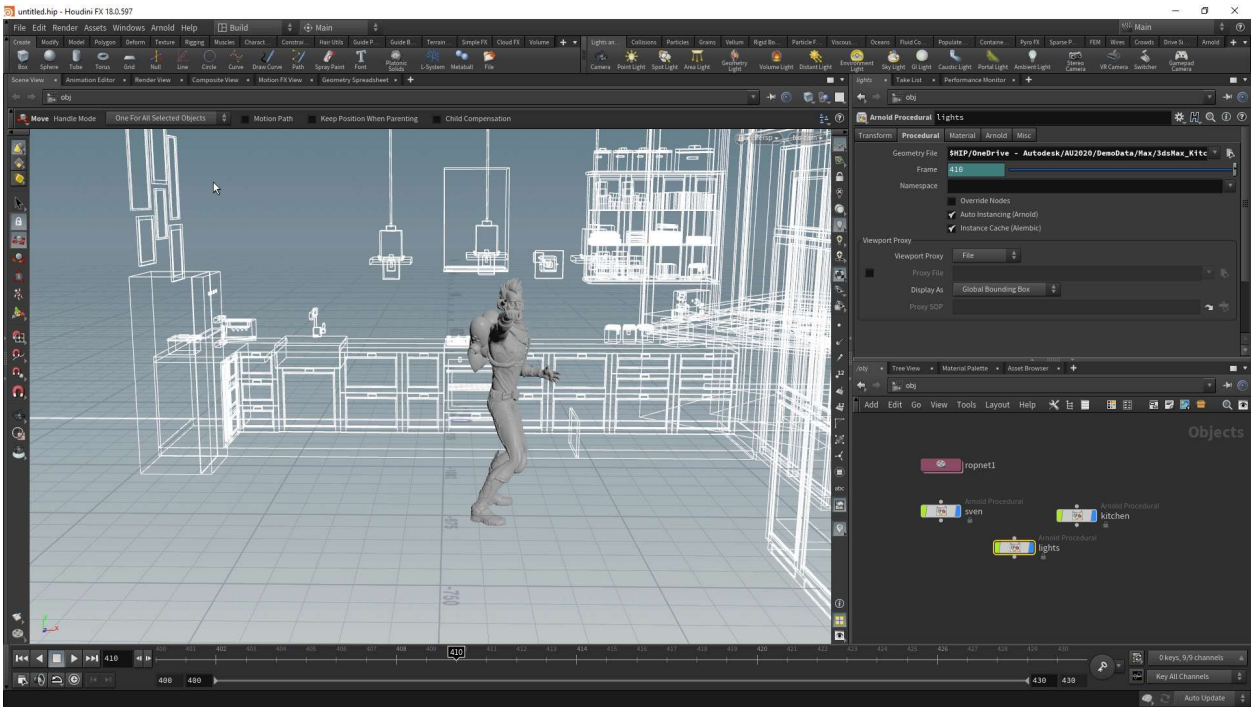
All assignments and changes that are done on the alembic file are contained under a look which is named default. A standIn node comes with a default look, but we have the ability to add new looks or edit existing look. We can create a new look from the existing look and make changes to it. In this case, we create a new look by duplicating the existing look. We change the suit material to be the red. Then we are able to switch between our look and have the render change between red and blue suit materials. This is a non-destructive way of making changes and edit to your look-dev scene.



We can export these look as as Arnold Operator graph with Arnold's own format or also as MaterialX. If we export it as an .ass file , we will need to bring them into the scene using the includeGraph operator to bind it to the geo again. We can bring the character with an animation into scene and very quickly apply the look using the includeGraph operator. We still retain the ability to switch between these look in the exported file , or assign new material . We have complete control over the look of this scene.

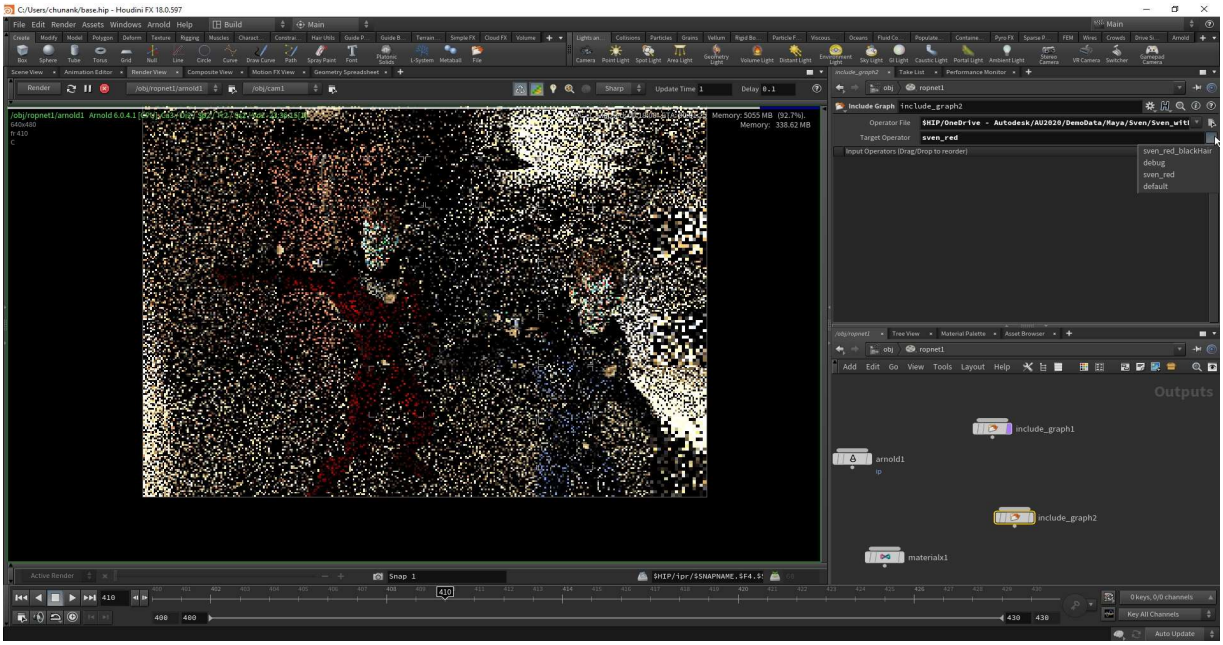
Arnold in Houdini (HtoA)

We import the environment from Max , and the character from Maya into Houdini. We also bring in the lighting setup which was exported from Max. We assign the looks for the environment which was a MaterialX file using the MaterialX operator in Houdini. We assign the character look using the include graph operator.



We'll need to do the operator assignments the houdini way. Once they are done, we can freely choose the look we want to apply to the character really late in the render stage if needed. We still have the ability to change the look and also make changes to the look.

We still retain the ability to set / toggle between the looks on the character. If for any reason we need to have a copy of the character with a different look, all we would need to do is bring a copy of the character into Houdini , assign the same look file and choose the look that is needed.



If there is a need for a look change , the scene that we have built is still live and nothing is lost. All we would need to do is export the new look from your look-dev scene and bring the update file into your Houdini scene. All of your new looks will show in here and we are able to assign that to the character. In the example below, we get a change just before render stage to make the hair of the red variant much darker and that was easy to do because our geometry is decoupled from our look and they are bound just before render stage.

This is the final image we render from Houdini. This image was produced using Max , Maya and Houdini with Arnold serving as the data driver between them.

Additional Resources

- MtoA Operators
- StandIn Tree Viewer
- HtoA Operators
- Interactive and flexible rendering workflows with Arnold Operators (featuring The Mill)
- Alembic, Operators, and MaterialX in MtoA 3.0.0

Connect with Us

- Arnold Answers