[2021]

[Rigging & Pipeline Tools]

[A FEW EXAMPLES OF TOOLS I WRITE]

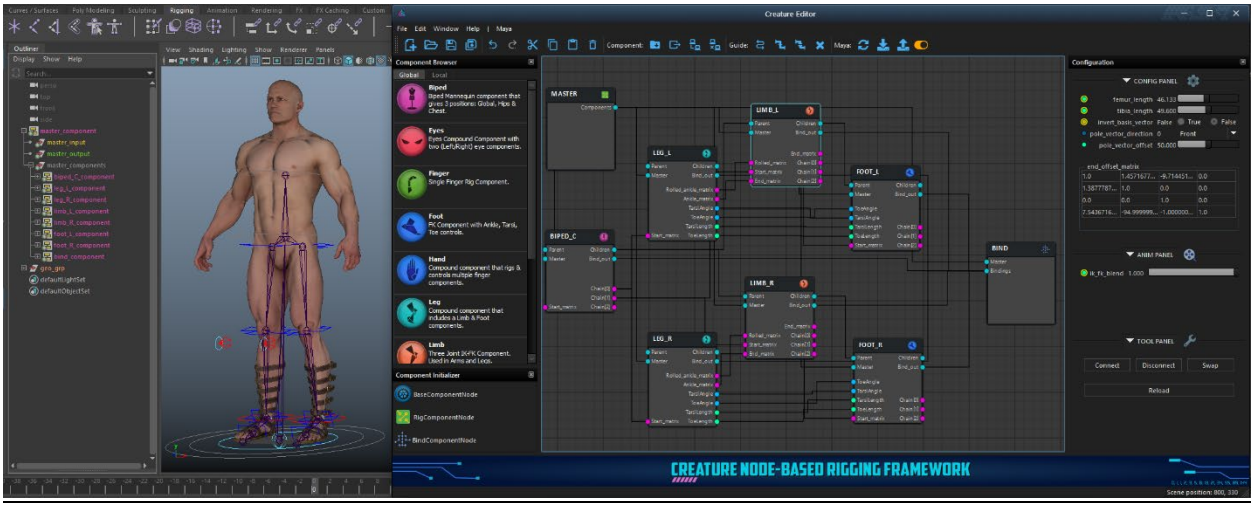
HARSHAD BARI

Creature Rigging Framework

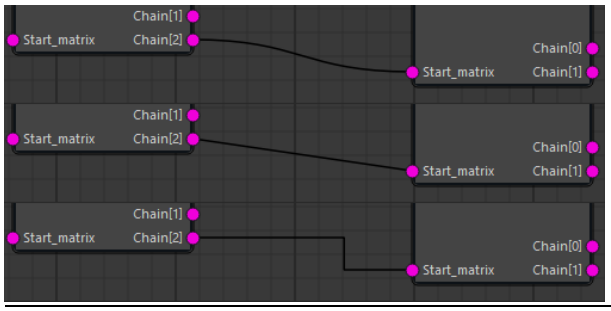
(Ongoing Personal project)

This project is an ambitious personal undertaking of mine to learn, create and achieve. I have been working on developing this in my free (non-studio-work) time for the past 2 years.

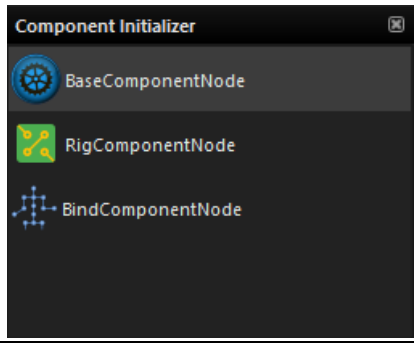
- Creature Rigging Framework is based on the component-based approach from CultOfRig. The Goal of my system is to help build structured modular components with set contract rules.
- The first prototype is completed. This milestone included writing an extensive core component class, a working visual editor, basic components and decoupled interaction with Maya with Qt Signals.
- Maya API is used extensively.



Edge Styles



Component Types



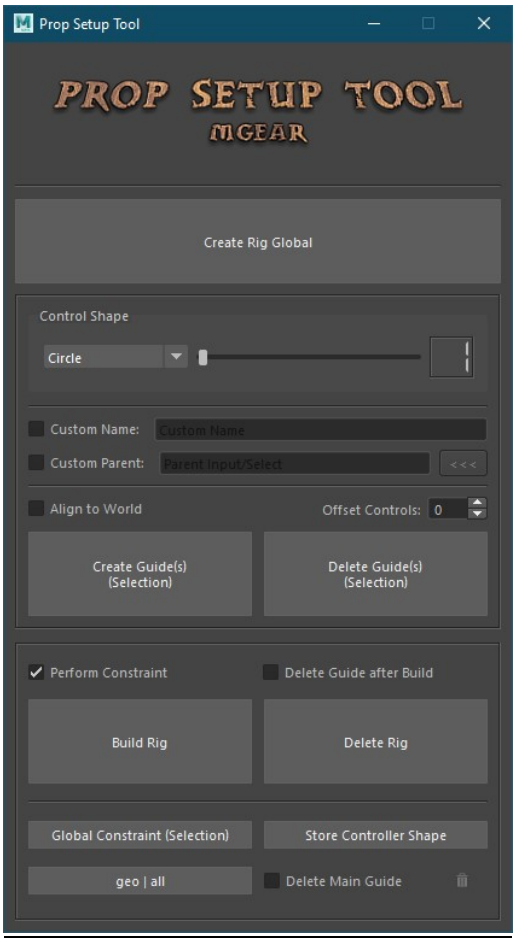
Prop Setup Tool (mGear)

Purpose:

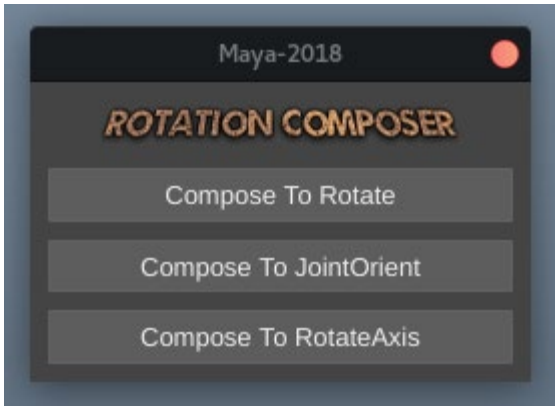
-- A Tool for environment artists to create Rigs for simple objects in a scene with controllers.

Features:

- Control shape & size selection
- Custom Parent Hierarchy and organization.
- Ability to Rebuild Rig (data is saved in scene to allow it).
- Create multiple offset controls.



Rotation Composer



Purpose:
-- Used for Joints mainly. Composes joint rotation to rotate, jointOrient or rotateAxis

Features:
-- Using Quaternion matrix of joint, extracts the world Rotation & sets it to desired attributes. (Zeroing others)
-- Very useful tool during rigging to move rotation values to rotate/rotateAxis/jointOrient without offsetting the joint.

Selection Handler

Purpose:
-- Store selections in PySide2 QWidget buttons.

Features:
-- Saves current selection for later use.
-- Button shows number of objects in the selection.
-- Works with both objects and components.
-- Tool supports add/subtract/toggle selection same as Maya default:
Ctrl: Subtract selection
Shift: Toggle Selection
Ctrl + Shift: Add Selection

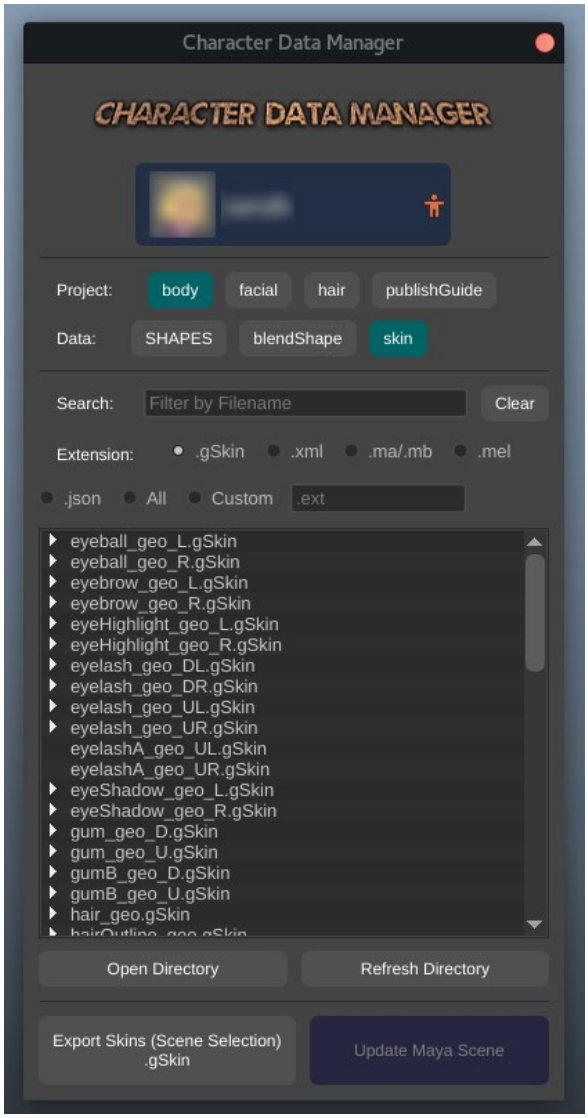
Character Data Manager

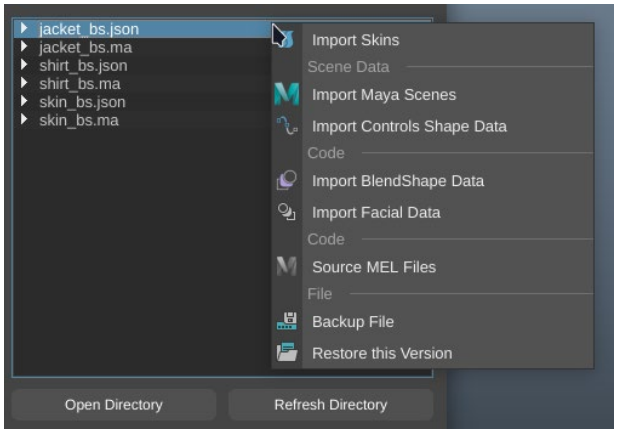
Purpose:

-- Manages character specific export data files (SkinCluster, Deformers, Shapes, MEL, JSON files) through a convenient UI.

Features:

-- Based on an existing version system, auto versioning for exported skin/deformer weight files.
-- Navigate Project structure easily using dynamically created buttons.
-- Filter files based on extension. Different extension allows context specific import export functions.
-- Filter search files.
-- Context menu for import or execute currently selected file(s).





CREATURE SKIN TOOLS

Purpose:

-- My Swiss-Army of Skin-Tools that I am continuously developing for fast personal workflow.

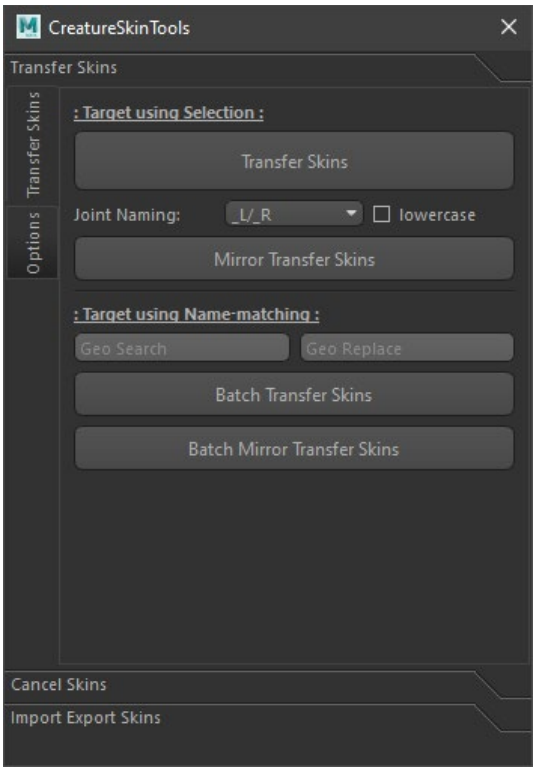
Features:

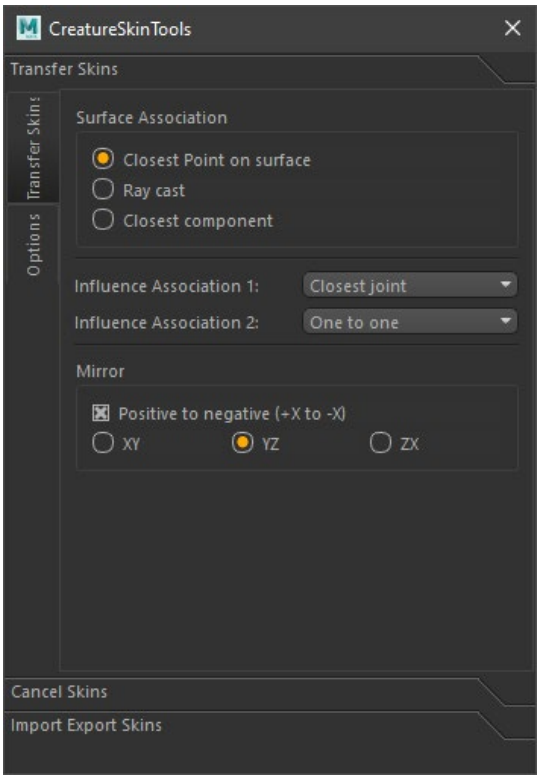
UI Features:

-- Consistent Tool Settings. Settings are exported to INI format and restored on initialization.
-- Focus on smart grouping of tools to increase speed of workflow.
-- Three Tabs separating TransferSkin, CancelSkin(BindPreMatrices) & Skin Exporter/Importer.

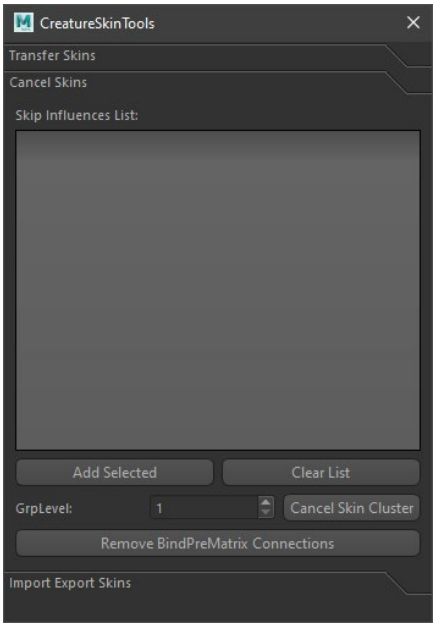
Transfer Skin:

-- Transfer SkinCluster from first selected object to all destination targets.
-- Joints list is queried from source and targets are automatically skinned to source joints.
-- Skinning can be transferred using name-matching (Search & Replace).
-- All above functions are supported with mirror name matching. E.g. Transfer skin from left to right.





Continue on next page…



Cancel Skin.

-- Add joints to UI List. Cancel Skin will connect joint parent's (user specified level) bindPreMatrix to skinCluster.

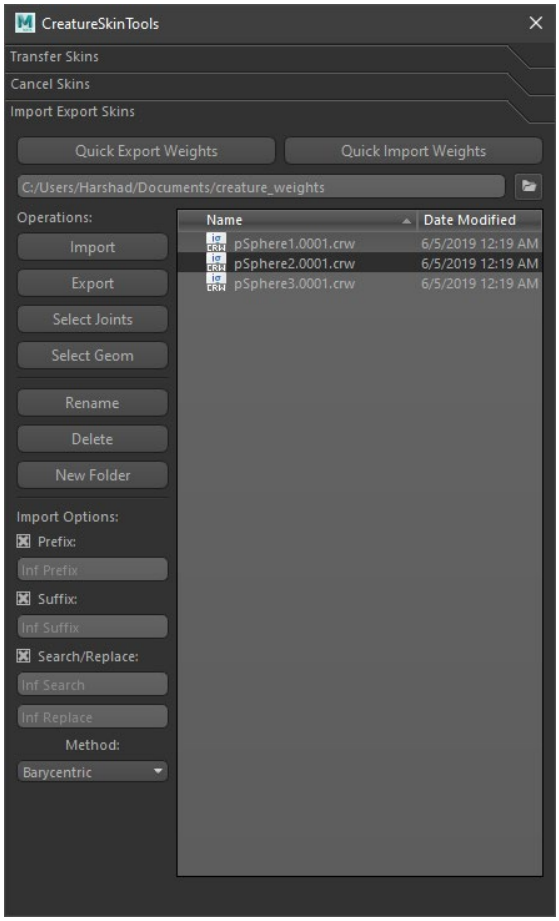
-- Convenient option to restore and remove all bindPreMatrix connections.

★ *Used for on-surface controls. E.g. Facial Rigging.*

Todo:

-- Split-pane UI showing all influences (left) and canceled influences(right).

Import Export Skin:



Features:

-- Core functions written in Maya Python API.

-- Exports skinfiles for all selected objects to separate files.

-- Select Geometry/Joints from skinFile.

-- Create/Rename folders and skin files.

-- Export skinfiles to specified directory.

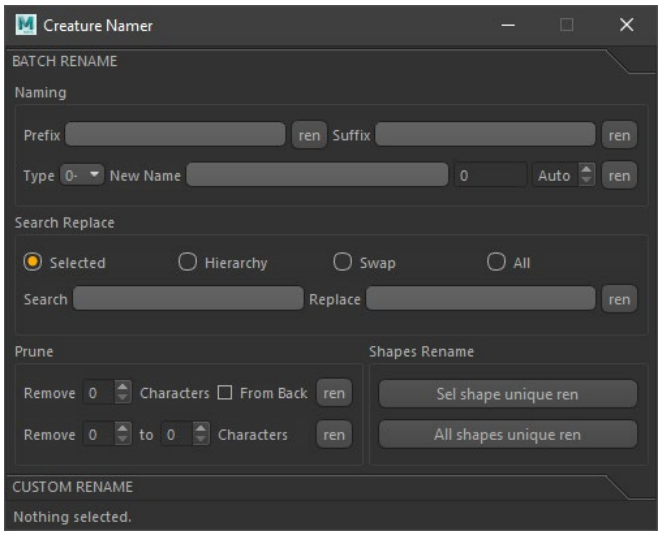
-- Ability to batch import all skin files from specified folder. #Organization

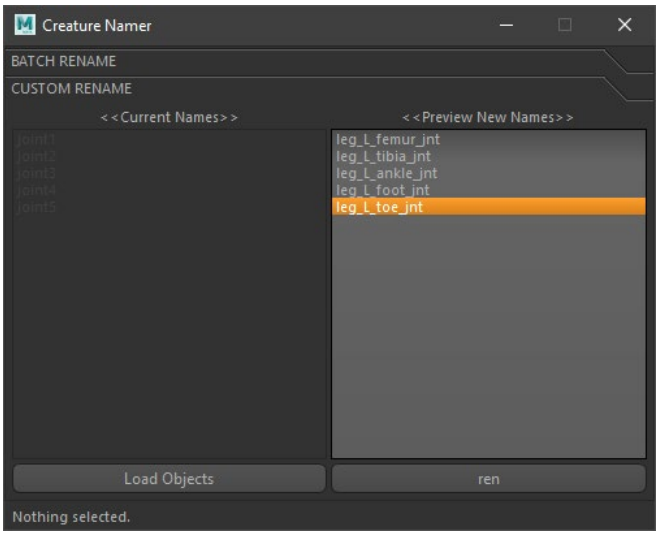
-- Import Skin to different joints based on influence name search/replace, prefix, suffix.

Todo:

-- Import skin weights in world space using barycentric coordinates.

CREATURE NAMER





Purpose:

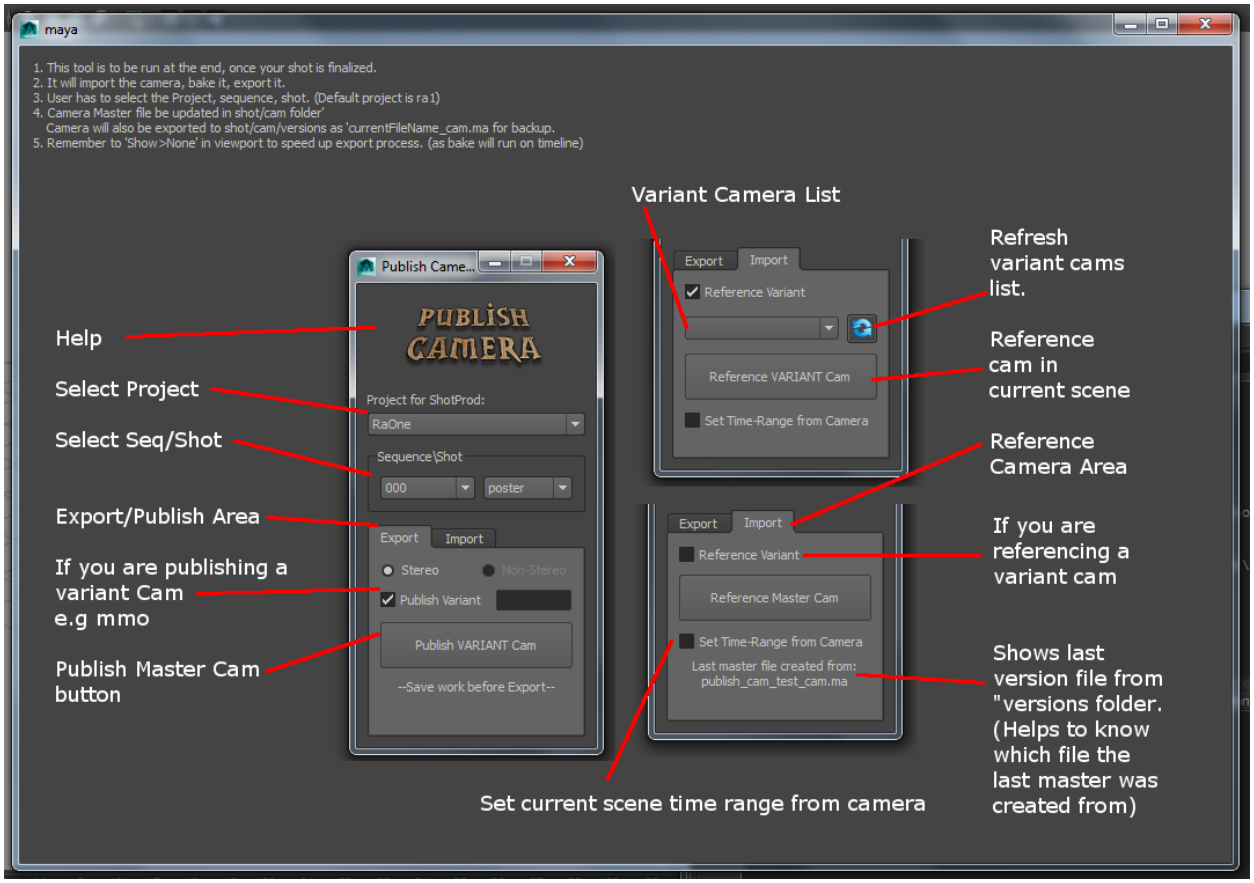
-- A tool to help fast rename objects with various options to improve Rigging time efficiency.
-- Custom rename a list of objects

Features:

Batch Rename:
-- Supports Prefix, Suffix, New Name with Auto Padding.
-- Search/Replace working with Selected, Hierarchy and All.
-- Swap Rename, useful for mirrored objects or prefix/suffix replace.

Custom Rename:
-- Load multiple objects to list and rename manually

PUBLISH CAMERA



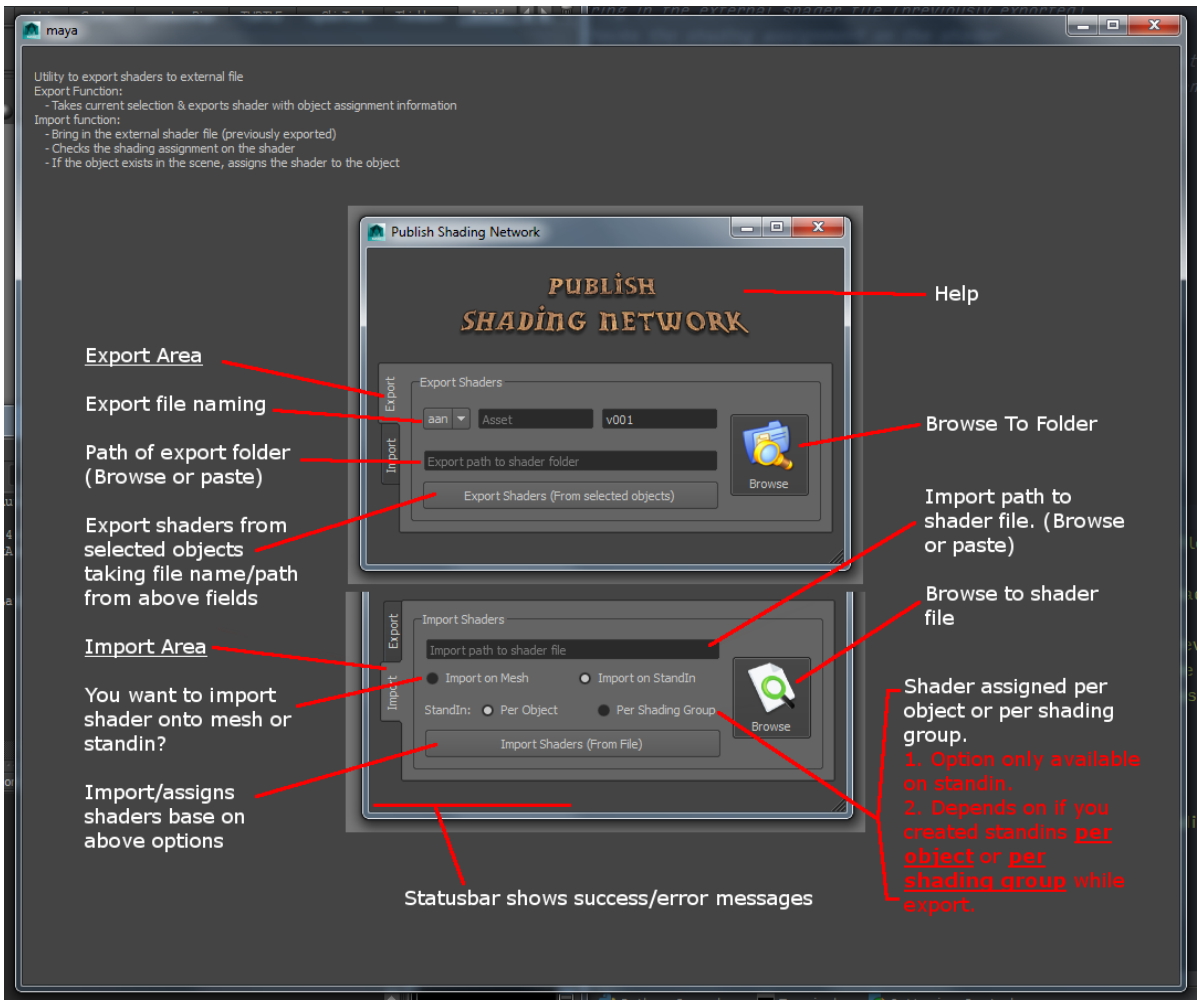
Purpose of this tool:

-- A tool to export/import Shot Camera stored on a central project directory.
-- User(Artist) has to select the Project, sequence, shot.

Features:

Exporter:
-- Camera will be exported to the specific shot/cam folder as "currentFileName_cam.ma"
-- Bake camera and export it with camera-specific information.
Importer:
-- Reference the master/variant camera based on user input
-- Set the scene time range from camera.

PUBLISH SHADING NETWORK



Purpose of this tool:

--A standalone tool to export/import Maya shading networks to an external file. (Path of this file can be set later at a studio level but right now its user-dependent).

--When exporting, the tool also saves object assignment to the shader node. So when importing, it will assign automatically to the objects (if object is present in scene).

--Options are described in the below image.

-- This tools also works well together with the create_ass_standins tool (see below)

Features:

Utility to export shaders to external file.

Export Function:

- Takes current selection & exports shader with object assignment information.

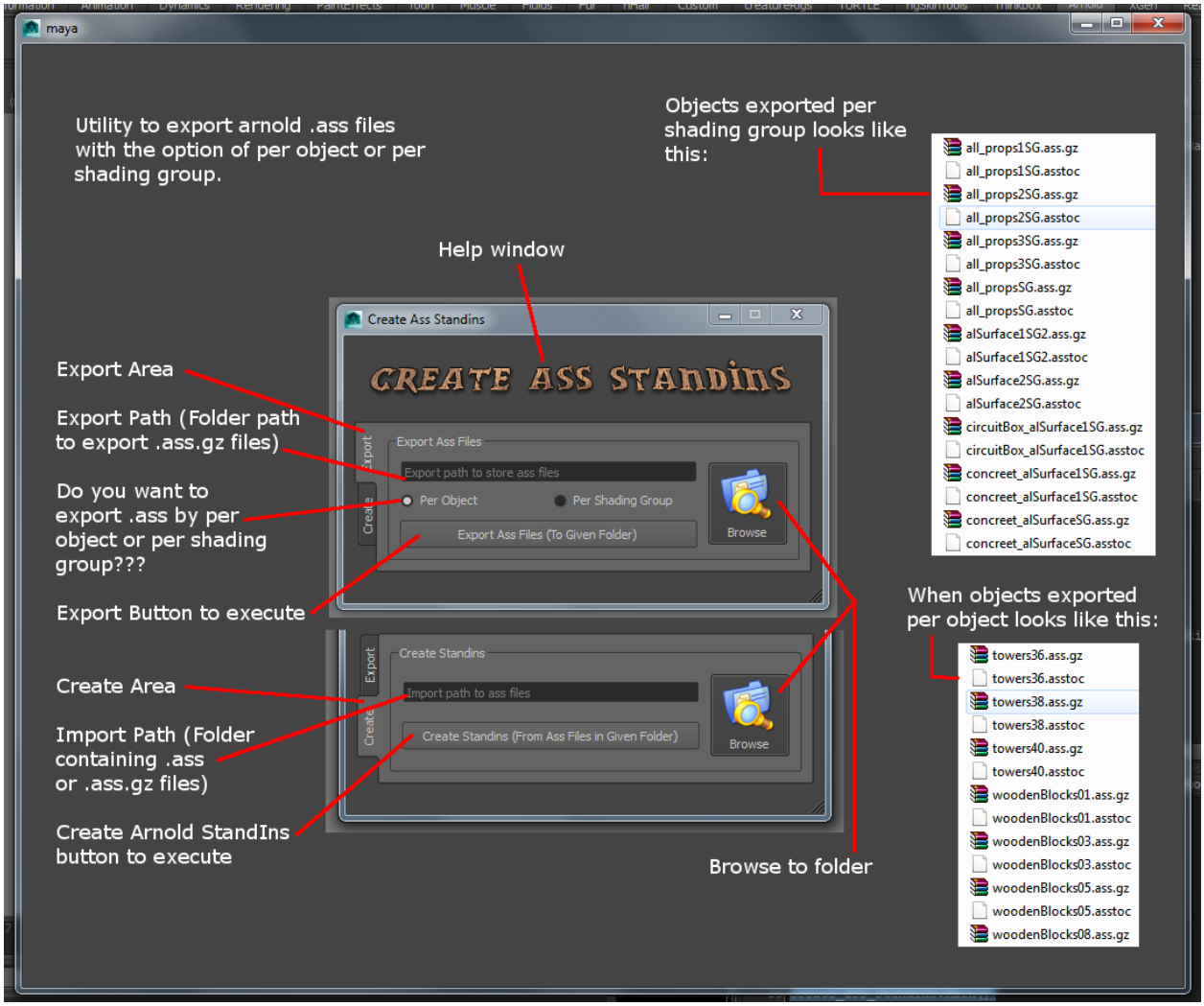
Import function:

- Bring in the external shader file (previously exported).
- Checks the shading assignment on the shader.
- If the object exists in the scene, assigns the shader to the object.

EXPORT/CREATE ARNOLD STANDINS (.ass)

Purpose:

Utility to export Geometry as Arnold stand-ins (.ass files)
-- Option of per object or per shading group.



CREATE ALEMBIC STANDINS

Purpose:
Given an Alembic file, the tool will read the contents and allow the user to create Arnold stand-ins in the current scene.

Features:
-- Read contents of Alembic file and display filtered Geometry results.
-- User can pin specific geometries from Alembic file to create stand-ins for it.
-- Support multiple alembic files.