



Character Production Pipeline

Created by Eric M. Scharf

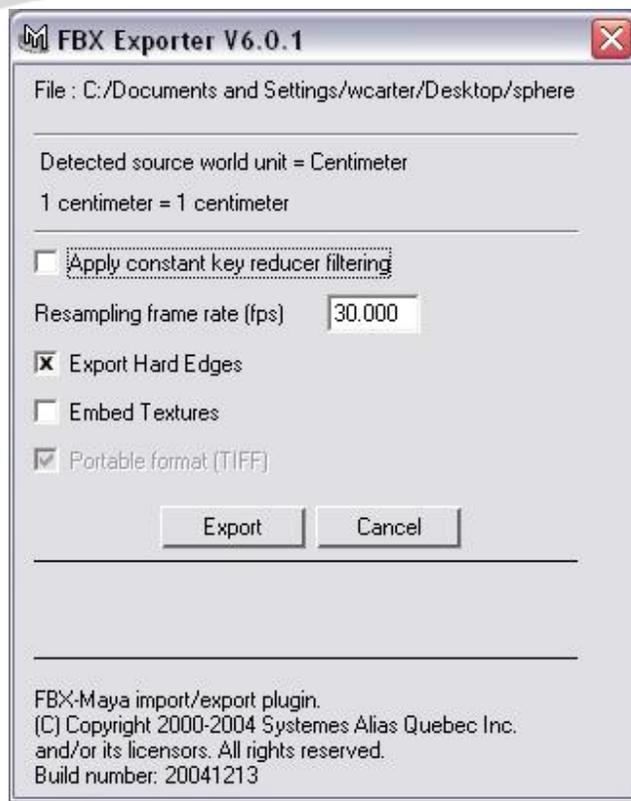
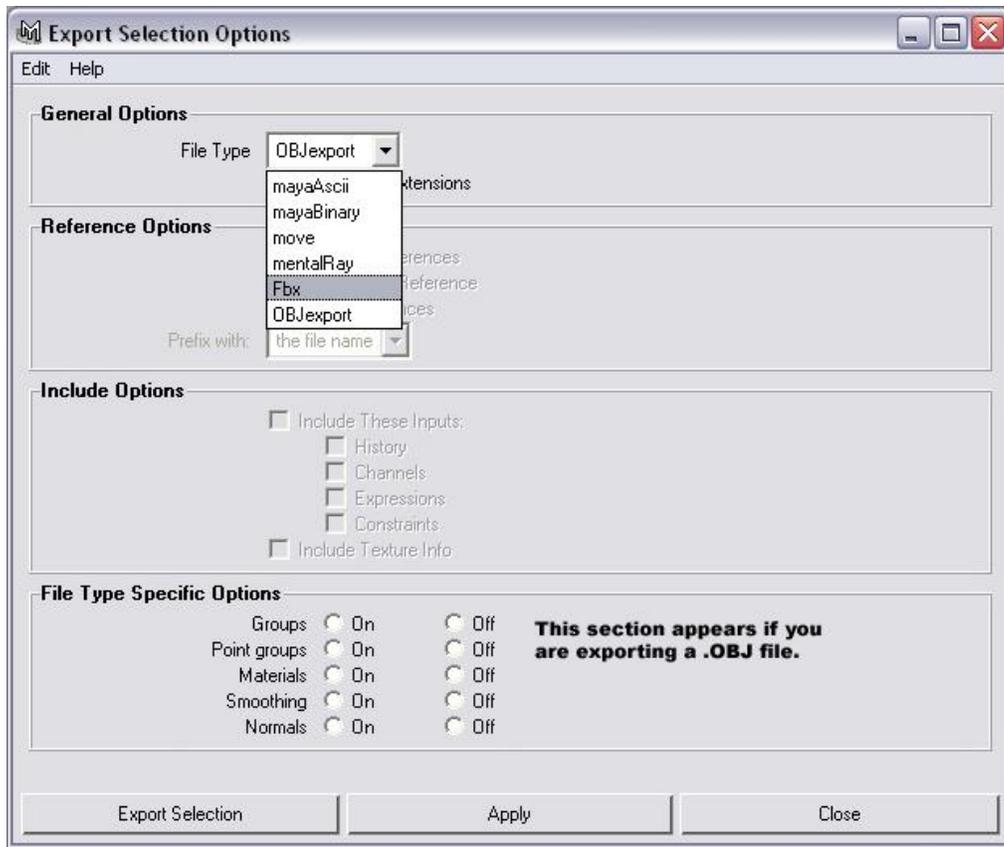
Created On: 04-03-2007

Last Modified: 06-25-2007

This outline displays the production pathway for generating a *default* character art asset (3D model, texture-map, UV layout, skeletal rig, node attachments, collision volumes, and exportation) for Relic Hunters (also known as Infinity Kids).

1. Creating Geometry (with Notes on Texture-Map Resolution)

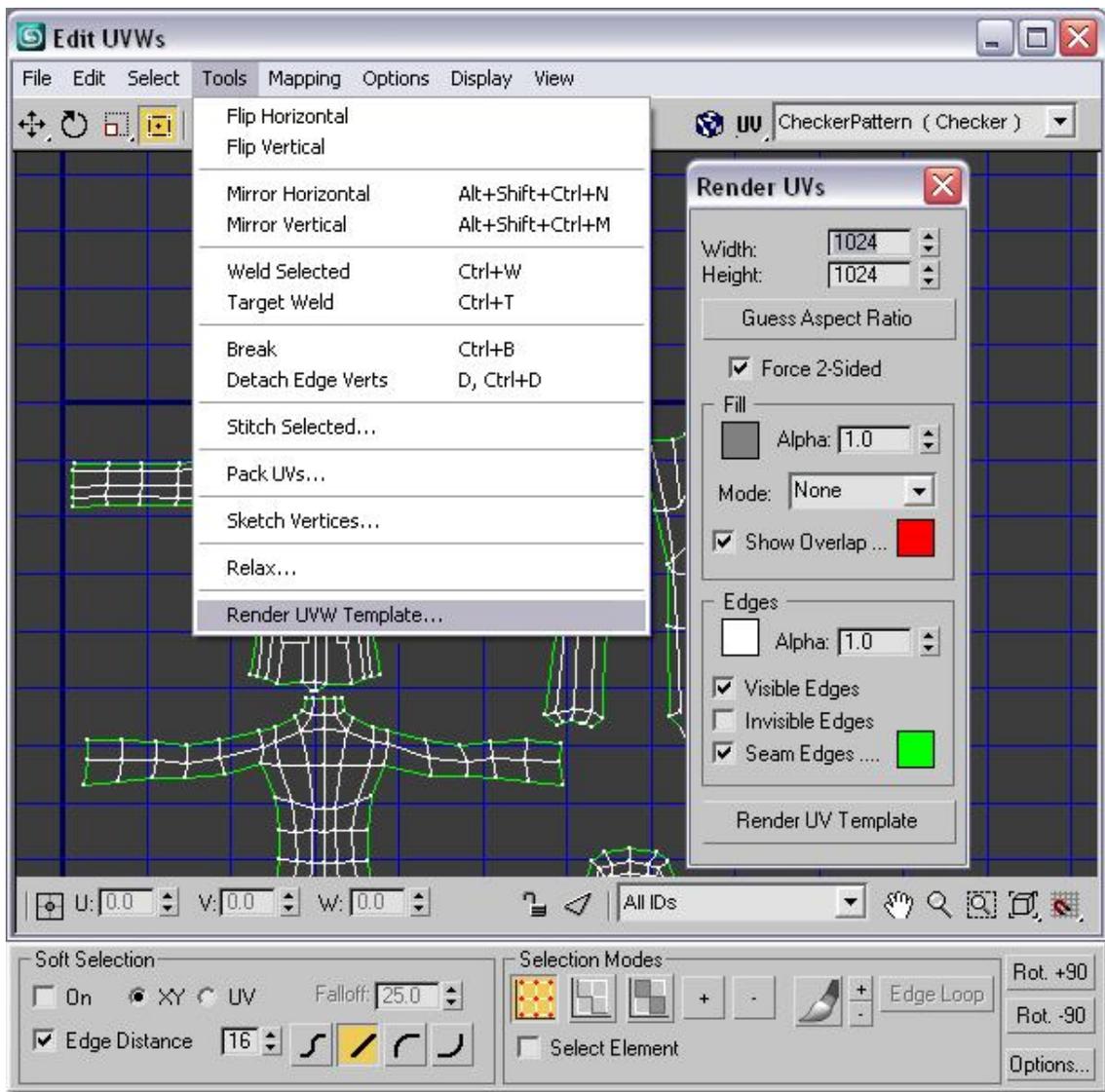
- All variations of the prototype character geometry will be built using both **3D Studio MAX** & **Maya** using primitive objects and editable polygons.
- **Each character will have an average polygon count between 2,500 to 4,000 polygons, with an additional 500 to 1,000 polygons specifically designated for gadgets and props.**
 - ***Please NOTE:*** The final camera zoom extents and geometry LOD quality will also affect the polygon counts of all geometry elements, regardless of type.
- **Each character will receive a single 1024x1024 SOURCE texture-map, in .PSD format, and that file will be converted into a 512x512 GAME READY texture-map (along with all other *potential* variants, such as normal, parallax, and specular), in .DDS format.**
 - The resolution of the GAME READY character texture-maps may be augmented from 512x512 if final camera zoom extents and frame rate dictate a change.
- **We will be using any of the ATI normal map generation utilities for normal map creation,** and they may be downloaded from: <http://mirror.ati.com/developer/index.html>.
 - [NormalMapper](#) generates normal (bump) maps from a high and low-resolution model, for PC and the APPLE platform.
 - [NormalMapperUI](#) - 3DS Max plug-in (version 1.22).
 - [NormalMapperUI](#) - Stand alone (version 1.21).
 - These executables can also be found on Topaz here: \\Server\projects\RelicHunters\ATI_NormalMapCreation_Utilities
- **Each character prop object / gadget will receive a single 1024x1024 SOURCE texture-map, in .PSD format, and that file will be converted into a 256x256 GAME READY texture-map (along with all other *potential* variants, such as normal, parallax, and specular), in .DDS format.**
 - The resolution of the GAME READY prop object texture-maps for may be augmented from 256x256 if final camera zoom extents and frame rate dictate a change.
- **We will be using a Photoshop plug-in to generate our .DDS-formatted texture-maps.** The plug-in, as well as some useful .DDS output "profiles," can be located on Topaz: \\Server\projects\RelicHunters\Art\Photoshop_DDS_Plugins
- The concept art images, upon which the Relic Hunters characters are based, are being created by Mike Phillips and Todd Kale.
- **Any and all 3d models generated within Maya are, then, imported into 3D Studio MAX as .FBX files using Film Box.**
- When exporting an art asset from Maya to 3D Studio MAX, make sure to turn **OFF Export Hard Edges**, within the properties of the export requester.



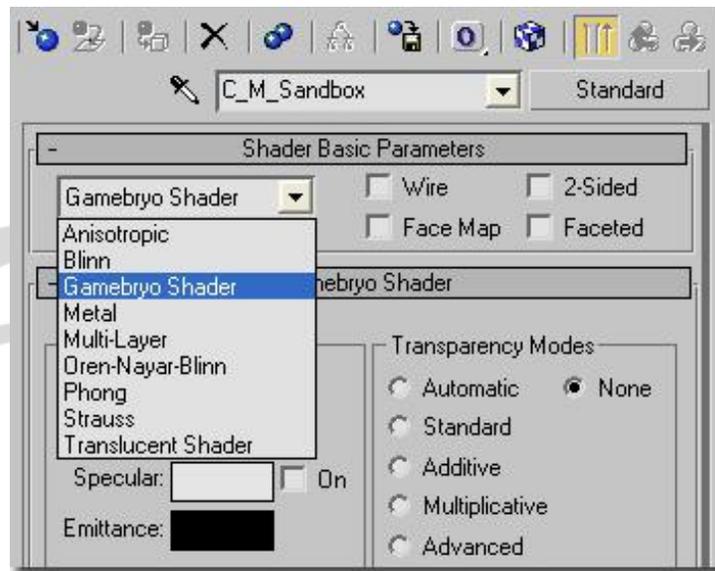
- The resultant export file will display in 3d Studio MAX with improper smoothing groups, which will need to be modified to match the 3D model display in Maya.
- Each character model is created as a fully-enclosed, single mesh.
- **The use of Z-Brush, Mud Box, or subdivision surfaces, in general, to assist in the creation of *higher detail* character models, will *only* be a *secondary, wish list* procedure.**
 - There are too many unknowns in the project schedule and with the available resources to take any other approach.

2. Creating Texture-Maps and UVW Coordinates

- The UVW coordinates are generated in 3D Studio MAX using the **UVW** modifier.
- Generating the UVW coordinates as a texture-map requires accessing the **Render UVs** requestor under the **Tools** tab. You access the **Tools** tab through the **Edit UVWs** window by applying the **Unwrap UVW** modifier to the character model.



- All SOURCE texture-map files are generated in Photoshop, in .PSD format, and, then, saved off in .DDS format for characters, props, and gadgets.
- Unless you are directly responsible for generating the UV coordinates and texture-map creation, you need to come to a fundamental, collaborative agreement with the texture-map painter(s) on the optimal layout for UV coordinates vs. how the texture-maps should be painted.
- Once an agreement has been reached on the priority of UV layouts vs. texture-map creation, you want to avoid having to apply any major alternations to the character model (followed by partitioning the model into several pieces and reconnection back into one solid piece), as the original UV coordinates and UVW layout will have been ruined.
- ALL UV coordinate maps are extracted at a maximum SOURCE art resolution of 1024x1024. The GAME READY versions of these files are reduced to 256 x 256 for gadgets / props and 512 x 512 for characters.
- The **Gamebryo shader** material, within 3D Studio MAX, needs to be assigned in the **Shader Basic Parameters** rollout.
 - Setting the specular property to white or grey further enhances the diffuse colors of a texture-map.



3. Creating and Weighting a Skeletal Rig

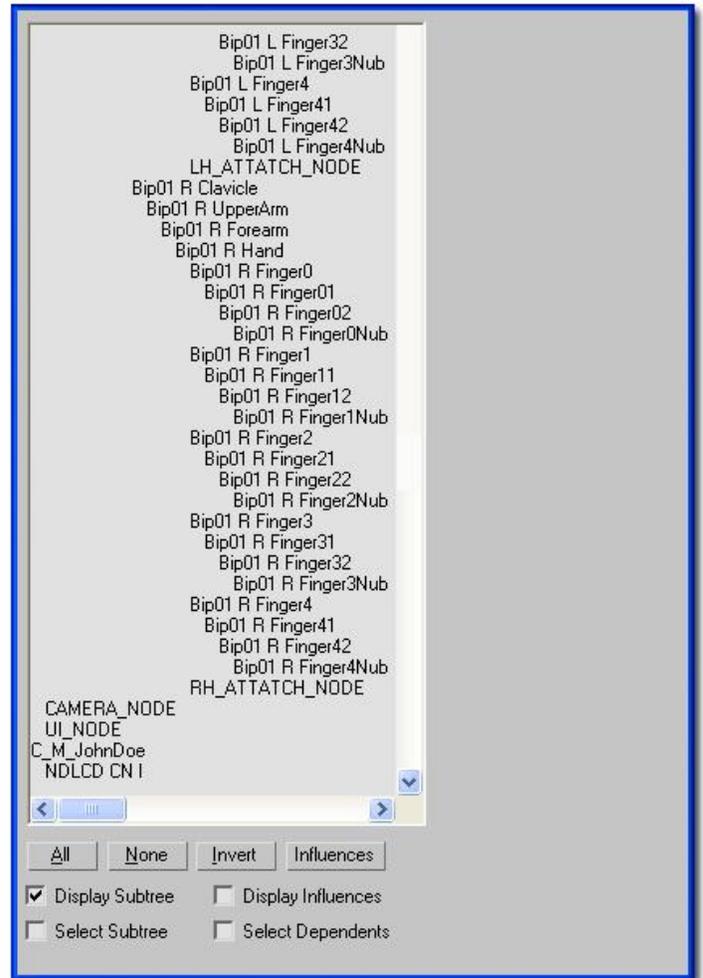
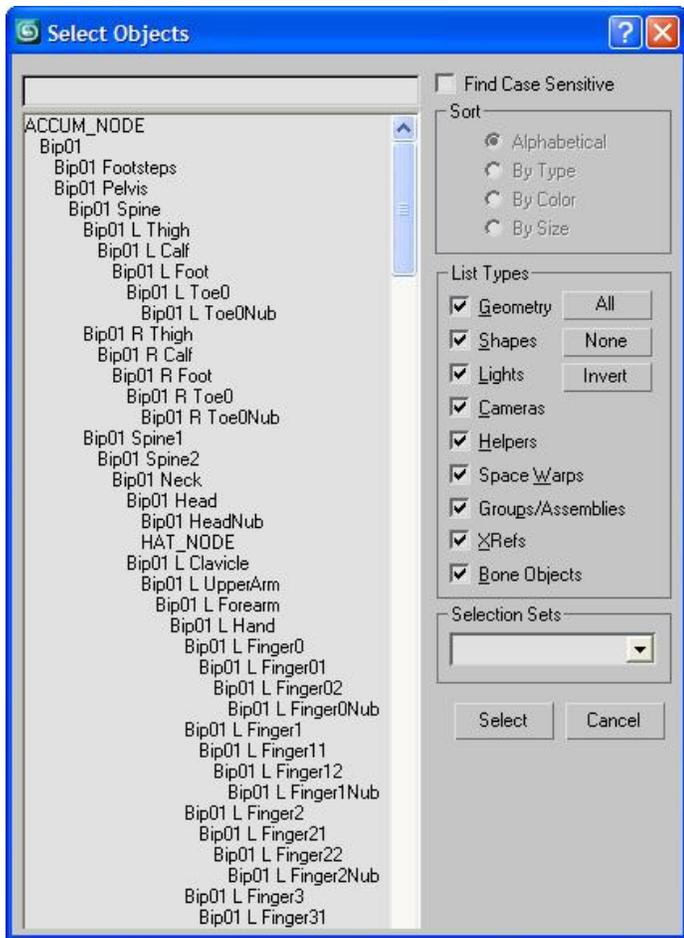
- **Character Studio** will be used to generate the character skeletons.
- Apply any and all rigging and vertex-weighting, to your character model and skeleton, using the 3D Studio MAX 9 **Skin modifier**.
- Envelope manipulation is straight forward, but vertex assignment can be tedious. The results, however, are better than that of the **Physique modifier**, *which is no longer supported*.
 - **Physique**, specifically, *never* allowed use of envelopes, did *not* provide any kind of **vertex-to-bone assignment chart**, and, generally, took *twice as long* as **Skin** to rig and weight a character.
- Do not proceed with skinning the character model until the UVW layout has been approved.
- Performing vertex / surface editing of the character model *after it has been skinned* will make the envelope and vertex assignments completely useless.

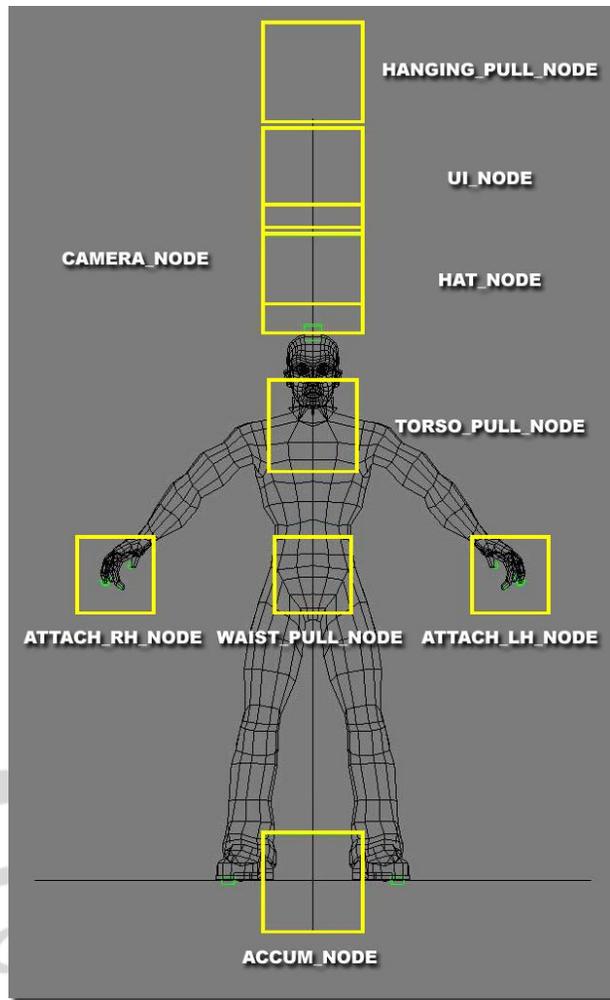
- The process of editing vertices on one side of the character model and using the mirror tool to paste those alterations into place on the opposite side of the character model is encouraged, and it will accelerate the editing process.
- Take a cautious approach to loading and saving skin envelopes, as the results can be hit-and-miss.

4. Setting up the Biped Character Rig for Nodes

A Biped should be set up using the following parameters for a **total bone count of 54** (with the hands receiving 30 of them). We will discover, in due course, whether or not 54 bones (with fully articulated fingers) is the appropriate number.

- A standard Biped rig will be established using these parameters, and it should subsequently be used as a base rig for re-proportioning new characters.
- We will be avoiding any animated props (of the permanently-affixed variety, such as a backpack), that would require adding bones to our Biped rig.
- The Object List, within 3D Studio MAX, for a 54-bone Biped displays like the following:





- **There should be a total of 9 nodes linked to the Biped:**
 - **ACCUM_NODE:**
 - This node should be set at 0-0-0, and Bip01 should be linked directly to the ACCUM_NODE (Accumulation Node).
 - **CAMERA_NODE:**
 - This node should have a pivot that is centered just above the top of the character head, and this node should be linked directly to Bip01.
 - **HAT_NODE:**
 - This node should have a pivot that is centered at the very top of the head bone, and this node should be linked directly to the head bone.
 - **HANGING_PULL_NODE:**
 - This node should have a pivot that is centered at the maximum height the character is supposed to *naturally* reach when jumping, and linked directly to the ACCUM_NODE.
 - This node is associated with any character animations where the character is hanging or sidling from a high surface.
 - **LH_ATTACH_NODE:**
 - This node should have a pivot that is centered appropriately on the left hand bone, and this node should be linked directly to the left hand bone.
 - **RH_ATTACH_NODE:**

- This node should have a pivot that is centered appropriately on the right hand bone, and this node should be linked directly to the right hand bone.
- **TORSO_PULL_NODE:**
 - This node should have a pivot that is centered at collarbone height, even with the tops of the shoulders, and linked directly to the ACCUM_NODE.
 - This node is associated with any character animations where the character pulls up from one surface to another surface, chin-up-style.
- **UI_NODE:**
 - This node should be a standard distance above the head bone, and linked directly to the ACCUM_NODE.
 - This node is associated with any UI element that is supposed to be on display in the 3D world space, including character health.
- **WAIST_PULL_NODE:**
 - This node should have a pivot that is centered at the abdomen, just above the tops of the hip bones, and linked directly to the ACCUM_NODE.
 - This node is associated with any character animations where the character pulls up from one surface to another surface,ommel horse-style.

PLEASE NOTE: Make sure that “NiOptimizeKeep” is set in the User Defined Properties for *ALL* attachment nodes.

The HAT and HAND attachment nodes should be adjusted for specific characters and props. **The UI and ACCUM nodes should *NOT* be relocated for consistency between character files.**

Established scale for characters should be 2 meters in height. All associated props should be scaled accordingly to match proportion and scale of the characters.

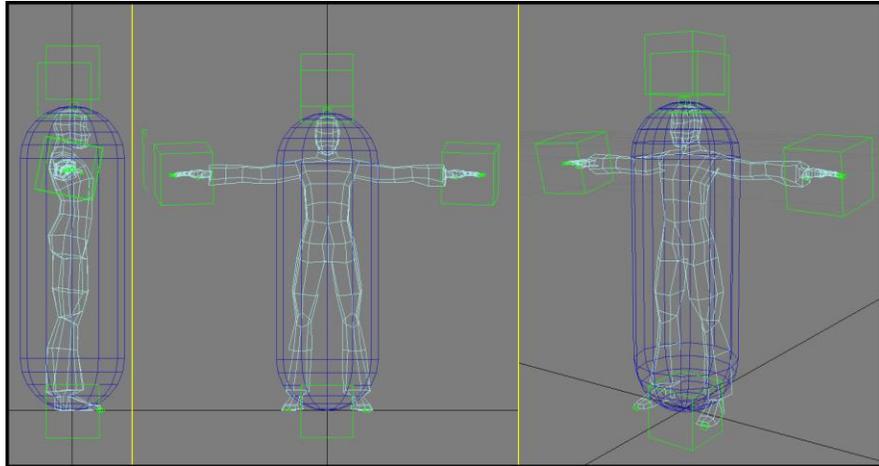
Prop Objects to-be-attached should be setup in the following manner:

- 1) Align the prop object to the attachment node.
- 2) Align the prop object pivot to the attachment node.
- 3) Center the prop object to 0-0-0 world coordinates.
- 4) Reset the XFORM for the prop object.
- 5) Collapse the modifier stack.
- 6) Export the prop object as an appropriately-named .NIF file.
- 7) Test prop object in the Relic Hunters Sandbox viewer.

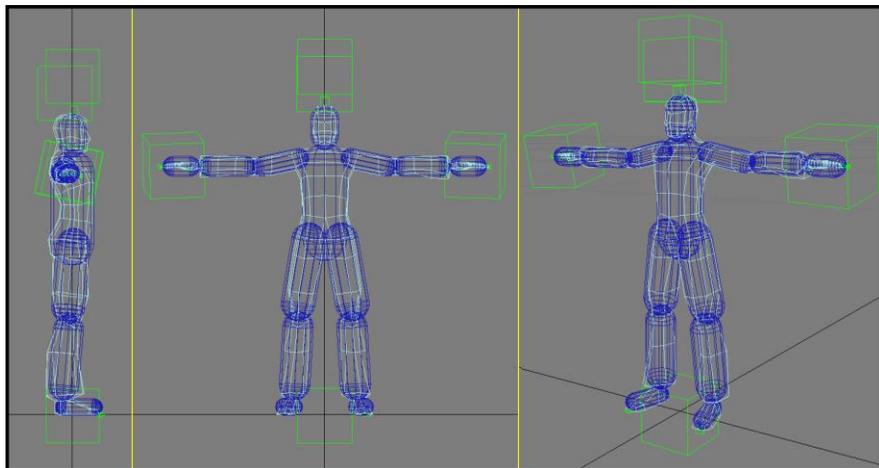
5. Creating Collision Volumes or Alternate Bounding Volumes (ABVs)

- There needs to be a collision volume or ABV for each character model in the game.
- The character ABV, for the Relic Hunters prototype, is a 3D Studio MAX “primitive” object that completely encases a character-at-rest (*not* in the standard T-pose).
- A single character ABV must always be named **NDLCD CN I**, and the ABV must contain *no* User Defined Properties.
- When a character contains *multiple* ABVs (SEE figure below), the ABVs must be named:
 - NDLCD CN I00, NDLCD CN I01, NDLCD CN I02, etc.

A single character ABV is being used for the Infinity Kids *prototype*. This capsule “primitive” ABV, rather than a more detailed collision volume, requires significantly fewer computations, which allows for a better frame rate in the game. *Accuracy* may suffer, however, depending upon your collision detection objectives.



A character model should ideally utilize a duplicate of its *own* geometry for “perfect” collision detection, however, this will hamper frame-rate. A happy medium, during full production, may involve multiple ABV capsules that account for the major components of a character model, thus, providing good speed and far more accurate collision detection.

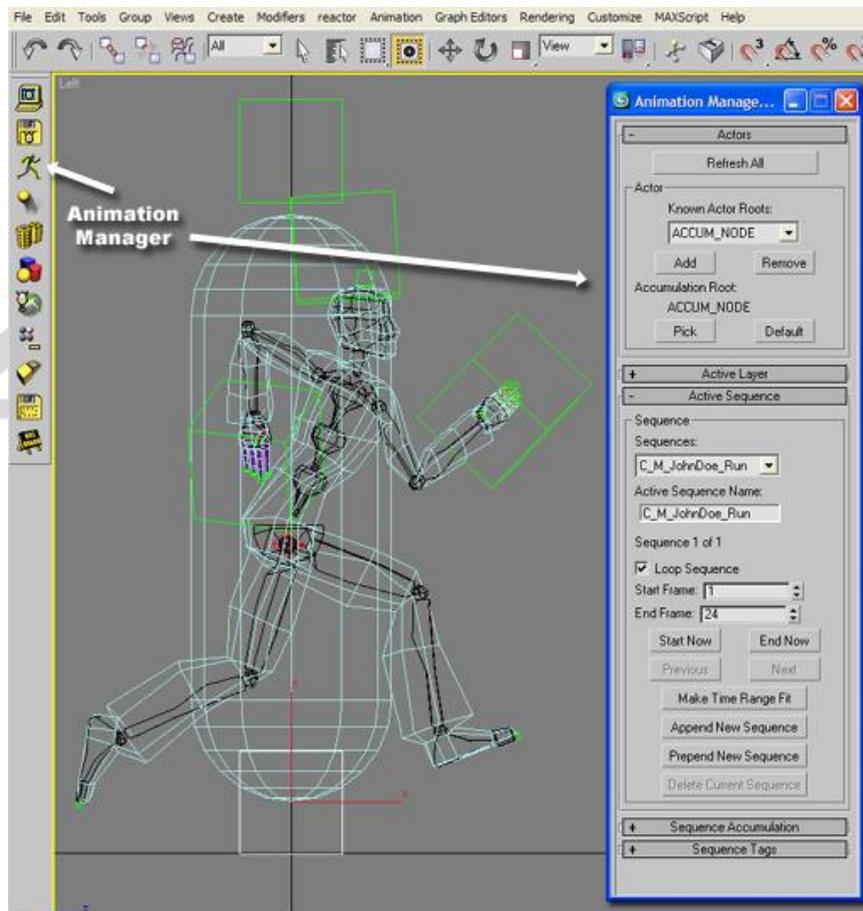


6. Exporting the Character Model Art Asset to Gamebryo

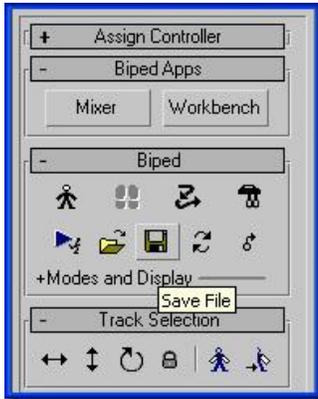
www.emscharf.com

- Gamebryo requires the creation of three unique export file types (.NIF, .KF, and .KFM), in order for any 3D Studio MAX-derived character art assets (3D model, texture-maps, and animations) to be successfully implanted in the game engine 3D world space.
 - You need to use the Gamebryo export plug-in, within 3D Studio MAX 9, to generate the .NIF and .KF files.
 - .NIF files always contain *only* the SOURCE 3D model and / or texture-map files (with the simple check of a box, within the export requester, you can include all associated texture-map files in with the 3D model).
 - The SOURCE 3D model, in this case, is a character, but it could be *any* category of 3D model (environment, vehicle, or prop), and the .NIF file would behave the same.
 - .KF files always contain *only* the SOURCE 3D animation, with a single .KF file containing a single animation.
 - .KFM files, generated using the Gamebryo stand-alone **Animation Tool**, act as a receptacle for .KF files. Each character will have a unique .KFM file, containing a series of .KF files associated with *that* character.
- In preparation for exportation of your character art asset as a .NIF file, you must make sure that **Bip 01** (a bone component from the biped skeleton within your character .MAX file) is linked to a **dummy object** (herein referred to as a **node**), containing the **NiOptimizeKeep** property.

- This node, labeled **ACCUM_NODE**, will act as your **accumulation root**, necessary for creating proper .KF and .KFM files for *ALL* character animation for Relic Hunters.
 - You need to make sure *ALL* nodes have the NiOptimizeKeep property applied to them.
 - You need to make sure that *ALL* nodes are linked to the top parent in the scene of your .MAX character file.
- ***Please NOTE:*** You can apply the NiOptimizeKeep property to any object within your .MAX file, either by typing the value directly into the **User Defined Properties** tab for any object, or, by the click of a button from the Gamebryo Toolbar, that can be loaded from within 3D Studio MAX.
- When exporting 3D art assets as .NIF files for Relic Hunters, regardless of art category, the associated texture-maps are to be saved as separate texture-map .NIF files.
- Before officially exporting your character *animation* asset, from within 3D Studio MAX, as a .KF file, you must make sure you have named your animation file correctly, within the Gamebryo **Animation Manager** plug-in.



- You must also assign a name to your animation using the **Active Sequence Name** area allows you to enter the correct name.
- Furthermore, be sure to set the **start** and **end** frames (matching the 3D Studio MAX timeline), as well as specify whether or not the animation is looping.
- You can now safely close the Animation Manager plug-in and save your character animation .MAX file, under the standard **File** menu using **Save** or **Save As**.
 - You can also **Save** and **Load** the *kinetic* animation file, through **Character Studio**, for future use with another character model art asset.

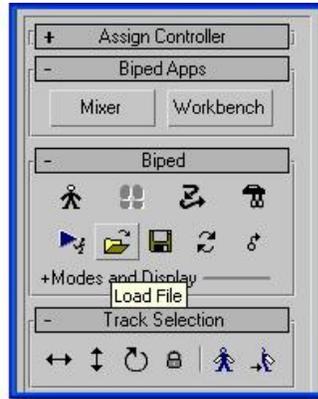


Select the bone called **Bip01**, and, under the **Motion** tab in 3D Studio MAX, click on the **Save File** button. Doing this will pull up the **Save As** requester displayed below.

Make sure you have **Save MAX Objects** checked, as this is the only way your nodes will be saved out along with your .BIP file.

Check **Save Segment at Current Position and Rotation** ONLY IF you need to save out a specific portion of your animation.

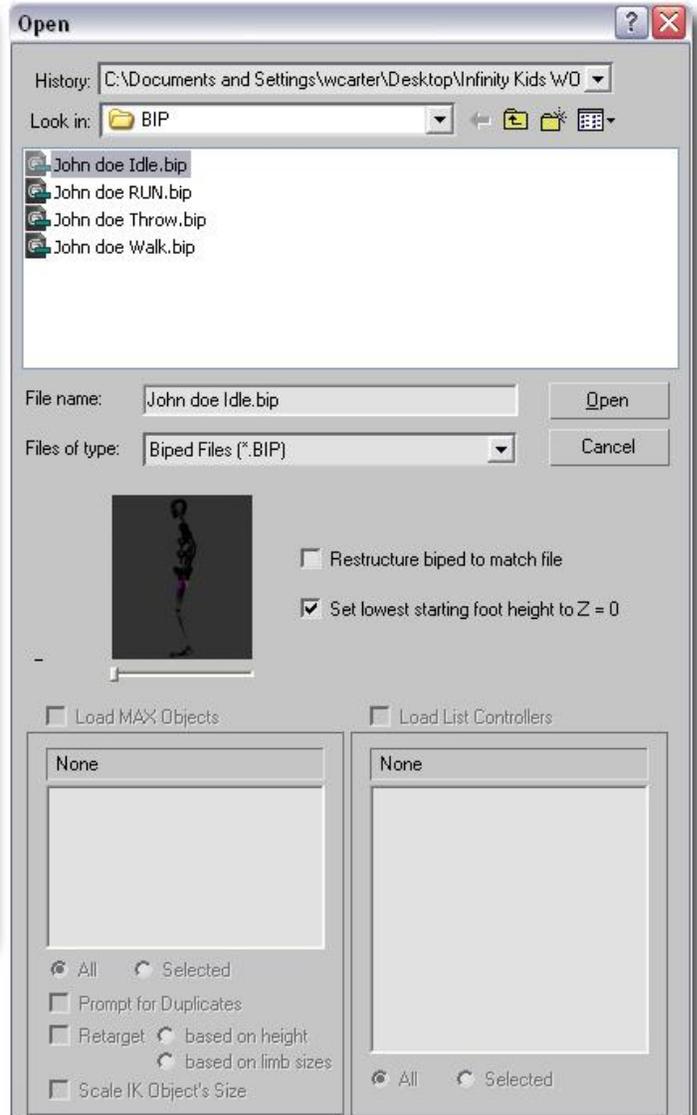
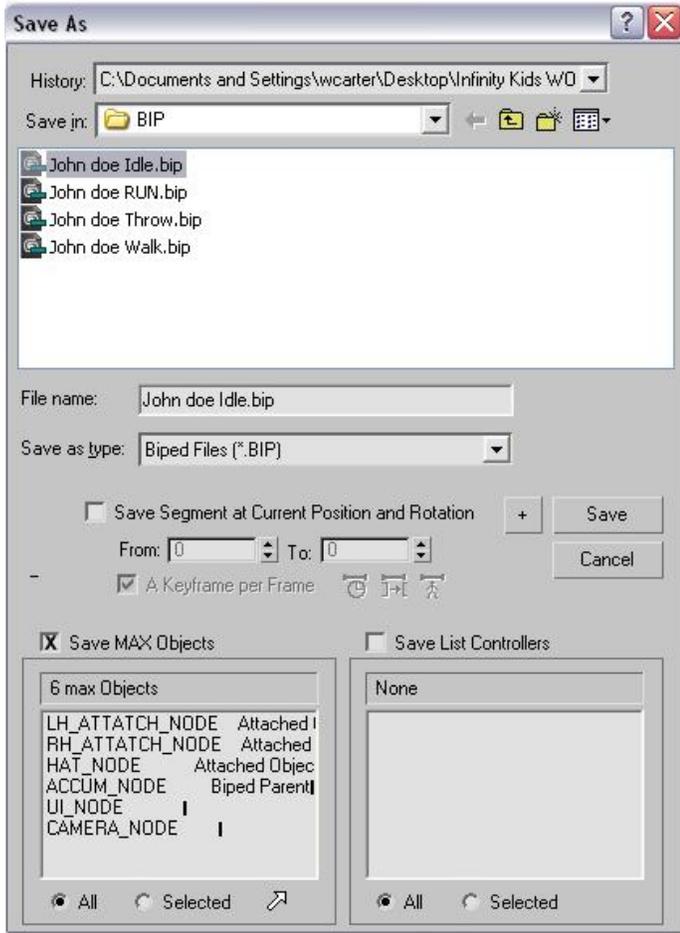
Otherwise, click the **Save** button.



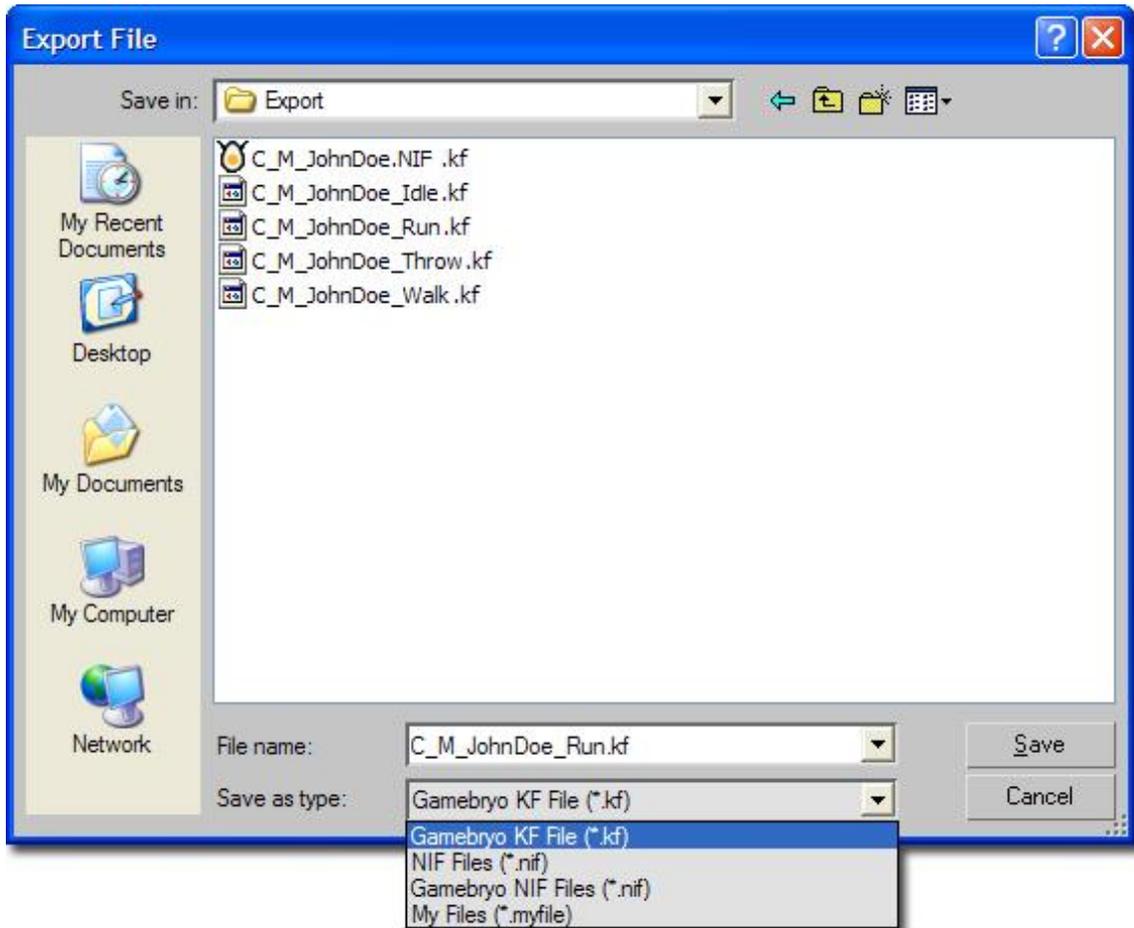
Select the bone called **Bip01**, and, under the **Motion** tab in 3D Studio MAX, click on the **Load File** button. Doing this will pull up the **Open** requester displayed below.

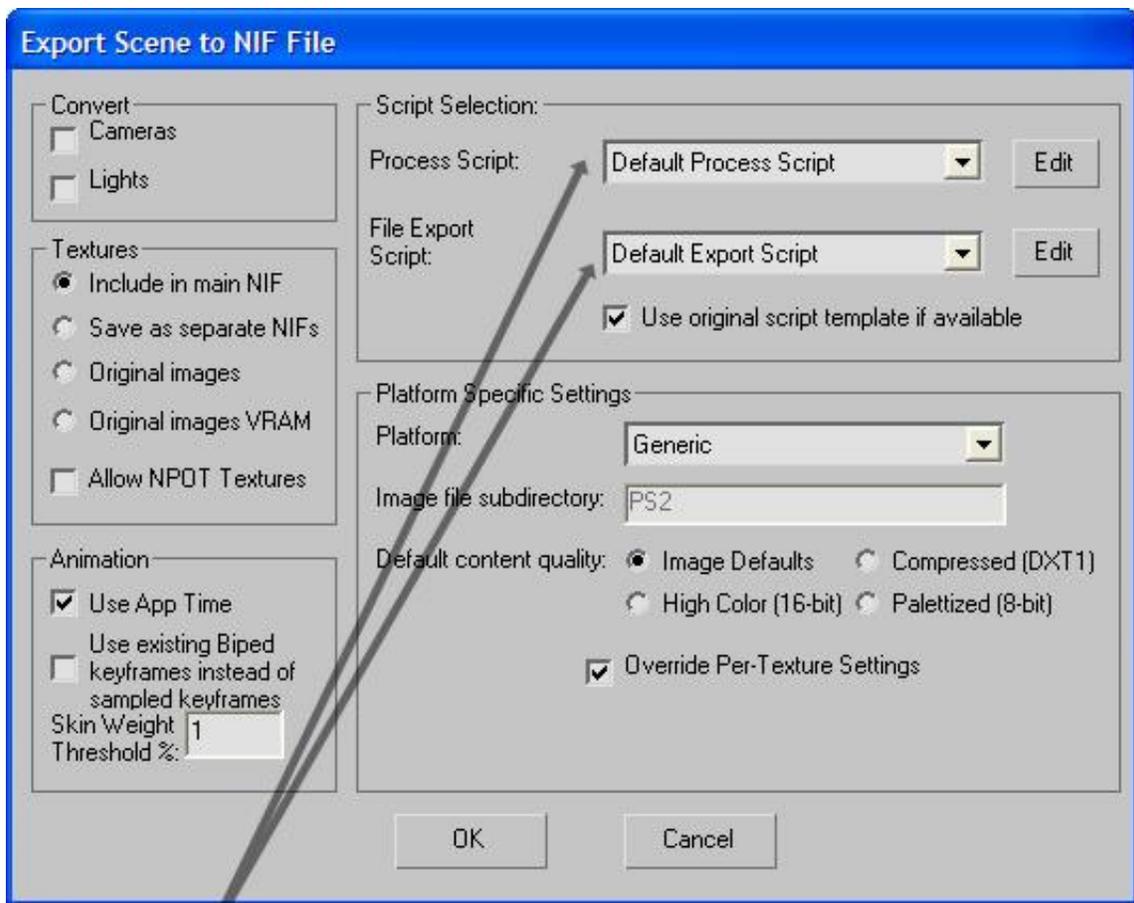
Make sure you have **Set lowest starting foot height to Z = 0** checked, and leave **Restructure biped to match file** unchecked.

Otherwise, click the **Open (Load)** button.



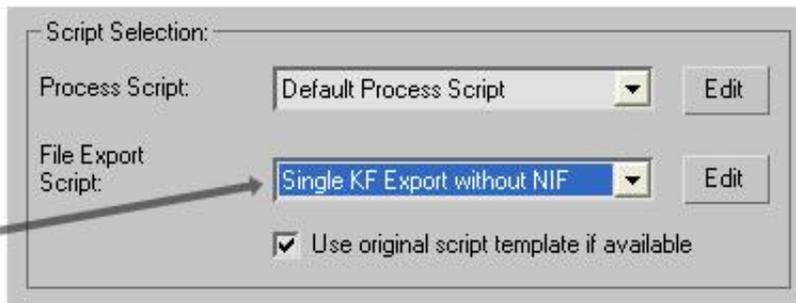
- ***Please NOTE:*** When you export your character file as a .NIF file, the Gamebryo exporter automatically generates a .KFM file as well. You must make sure the character file you are attempting to export as a .NIF file contains no animation.
- The Gamebryo **Export plug-in** requester is displayed below for both .NIF and .KF files.



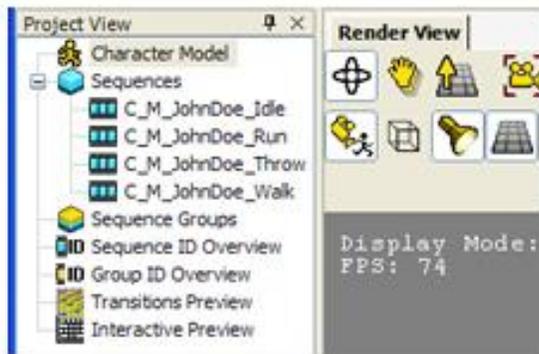
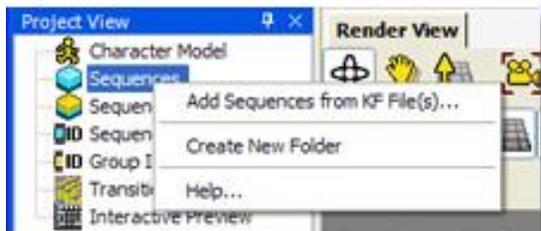
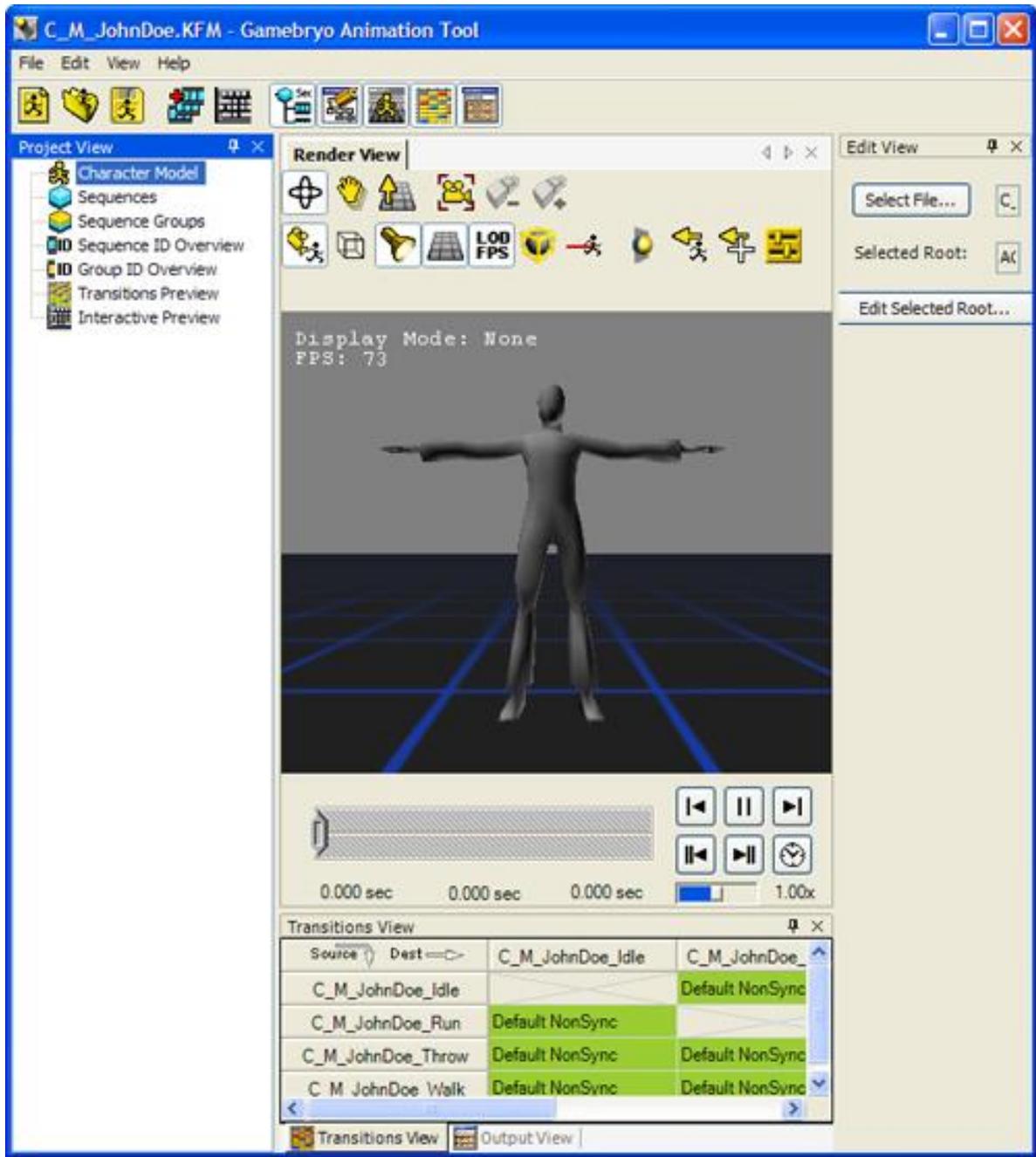


When exporting a .NIF file, the Process and Export Scripts should remain as **Default**.

When exporting a .KF file, the Export Script should be changed to **Single KF Export without NIF**.



- When the .NIF file is created, along with the automatically-generated .KFM file, you need to open up *that* .KFM file within the Gamebryo stand-alone **Animation Tool**. Your static character 3D model will be displayed in the standard T-pose.
 - You need to right-click on the **Sequences** tab, within the left-and-side **Project View** panel.
 - Then, select **Add Sequence from KF Files**. This function will place your newly exported .KF file into the **Sequences** list.
 - You will ultimately be depositing *ALL* of a specific character's animation sequences (or .KF files) into the Sequences list for that character's .KFM.
- Upon completion of adding all necessary .KF files, go to the File menu and click **Save** or **Save As**.



7. Creating and Exporting Prop Objects (General Props and Gadgets)

A complete list of required prop objects and gadgets, for the prototype, and, ultimately, for full production, is still being developed by Design. Once this list has been made available to Art and Programming, to collectively determine how the associated art assets and code infrastructure need to be generated, then, work can begin in earnest on this category of art assets.

- Prop objects and gadgets, as previously stated, will receive 500 to 1,000 polygons, depending upon priority level.
- Prop objects, whether static or animated, follow the same export rules as character models do.
- Regarding prop object sizes, the current agreement between all project disciplines is that general prop objects (not gadgets) will range in size from ½ meter to 1 meter, and no other sizes in-between or beyond.
 - This will allow Art to generate *only* two unique sets of “pick up object” animations, rather than an incredible amount designed to support all types of objects.
- Regarding node usage, the current agreement between all project disciplines is that each interactive object in the 3D world space will require “mount” nodes, or triggers, for where a character would begin to interact with a given prop object.

8. Checking the Character Model Art Asset into Perforce

- Once you have verified that your exported character art asset is error-free, you need to check your art asset (as well as all associated SOURCE art files) into Perforce.
- **There are three Perforce directory locations, as of 06-25-2007, where you are obligated to check in your environment files.** The prototype main character, Basil, for example, requires that you deposit files into these locations on your local system:
 - **C:\Clients\Project\Art\Characters\Basil\Export**
 - Deposit all .NIF, .KF, .KFM, and .DDS files (*even though* the .DDS files are included with the associated .NIF files) here.
 - **C:\Clients\Project\Art\Characters\Basil\Source**
 - Deposit all .MAX, .BIP, .PSD, and .DDS files (as the .MAX file map paths point to the .DDS files, not the .PSD files) here.
 - **C:\Clients\Project\Testbeds\Sandbox\Data\Objects**
 - Deposit all .NIF files here.
 - Everyone on the Relic Hunters team, until further notice, is running the game from the Sandbox executable, thus, it is extremely important to make sure all of your .NIF files are up-to-date in this directory above all else.