

# **Uncharted 2 Character Pipeline:**

**An in-depth look at the creation  
of U2's characters.**

RD



# Who Are We?

**Rich Diamant**

*Lead Character Artist*

**Judd Simantov**

*Lead Character TD*

RD



# Presentation Overview:

- Pipeline & Process Overview
- What we wanted to fix from Uncharted 1
- 
- New challenges specific to Uncharted 2
- Outsourcing
- Conclusion
- Questions & Answers

# Modeling Process:

- **Base Sculpt Mesh**
- **Arbitrary Game Mesh**
- **Texturing/Sampling**
- **Shader Setup**

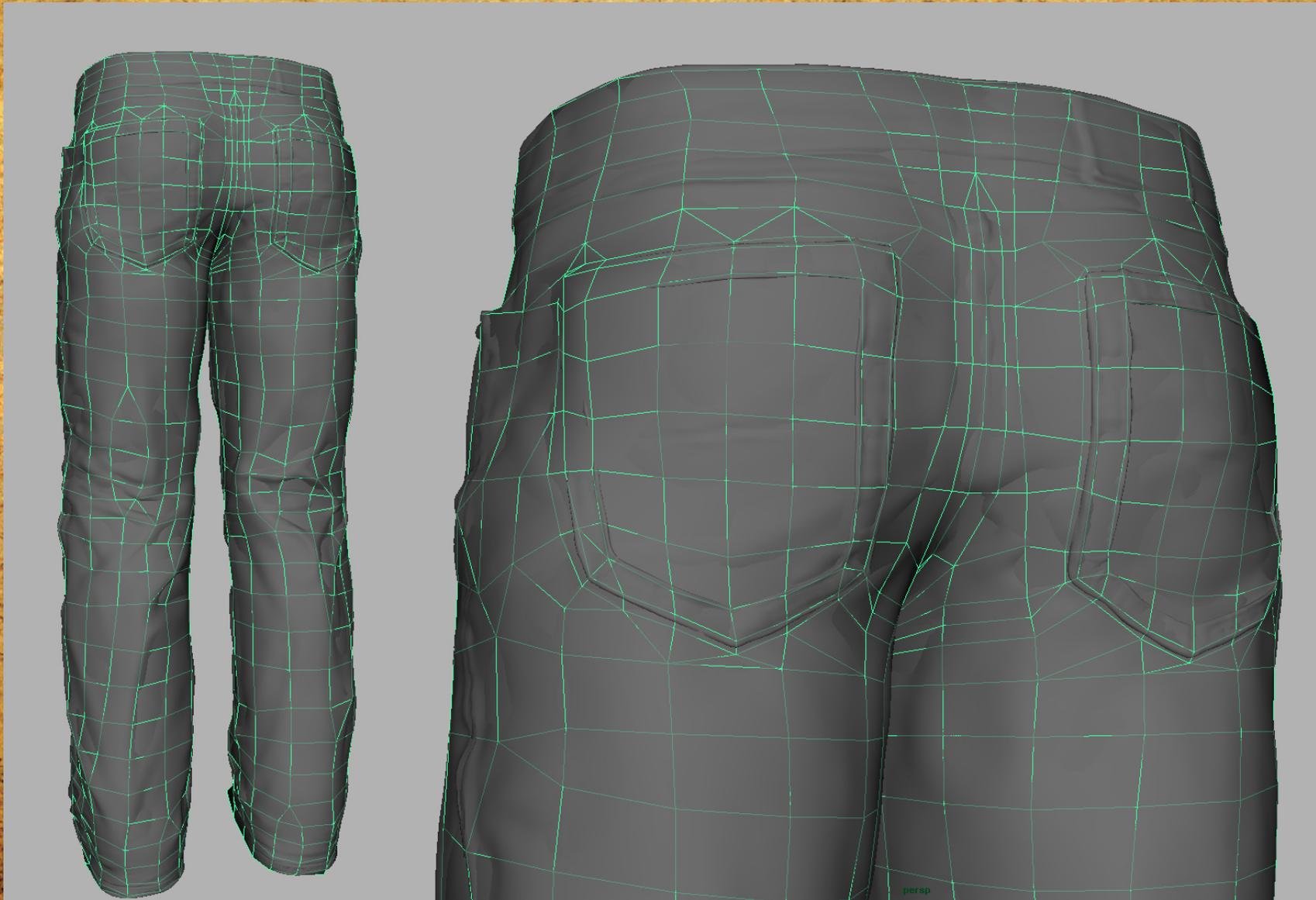
# Base Sculpt Mesh:

- Good Topology for sculpting
- Uv's for texturing



# Arbitrary Game Mesh:

- Game Resolution Topology (As low as possible)
- Correct Uv's for texturing



# Arbitrary Game Mesh:

Game Mesh



Base Sculpt Mesh

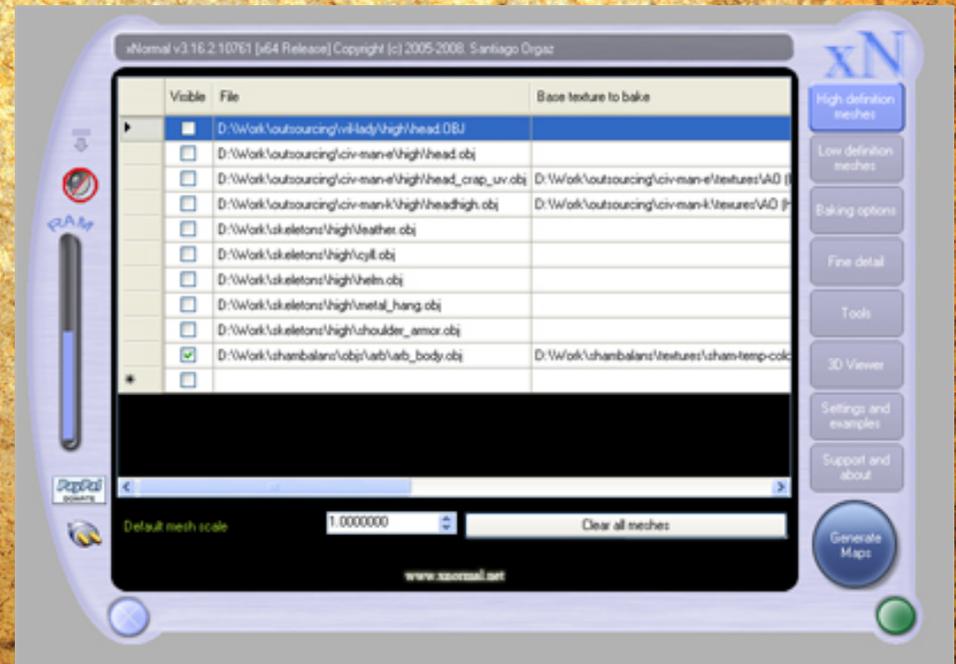
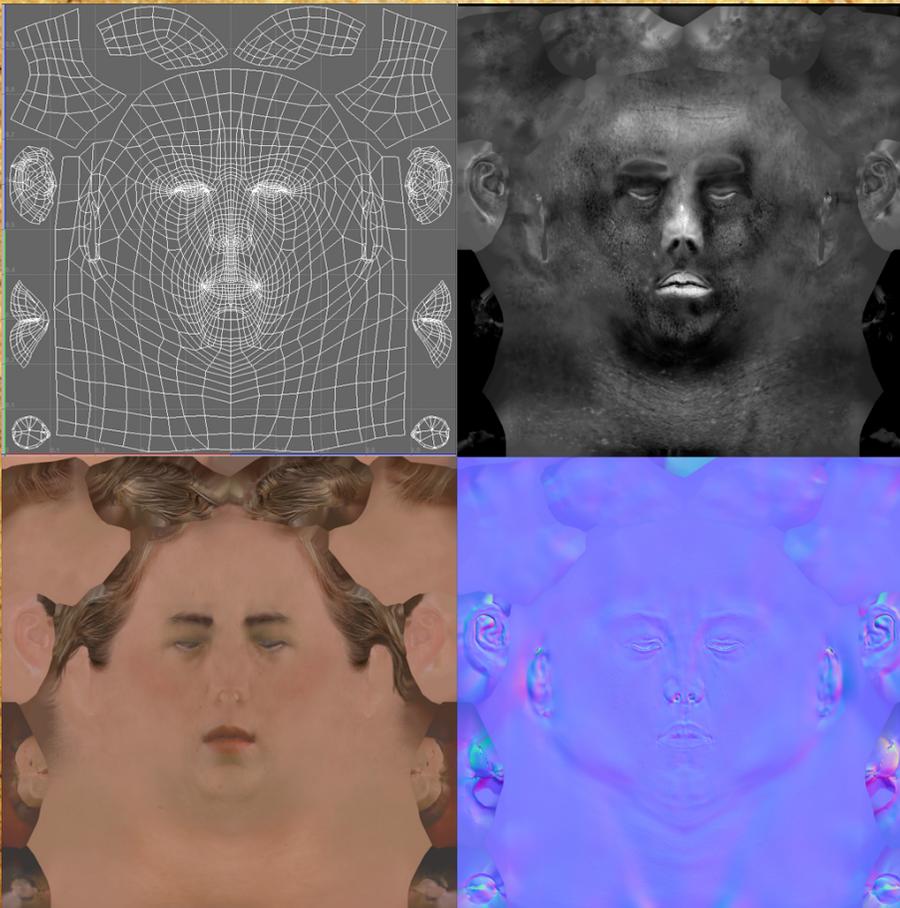


High Res Sculpt



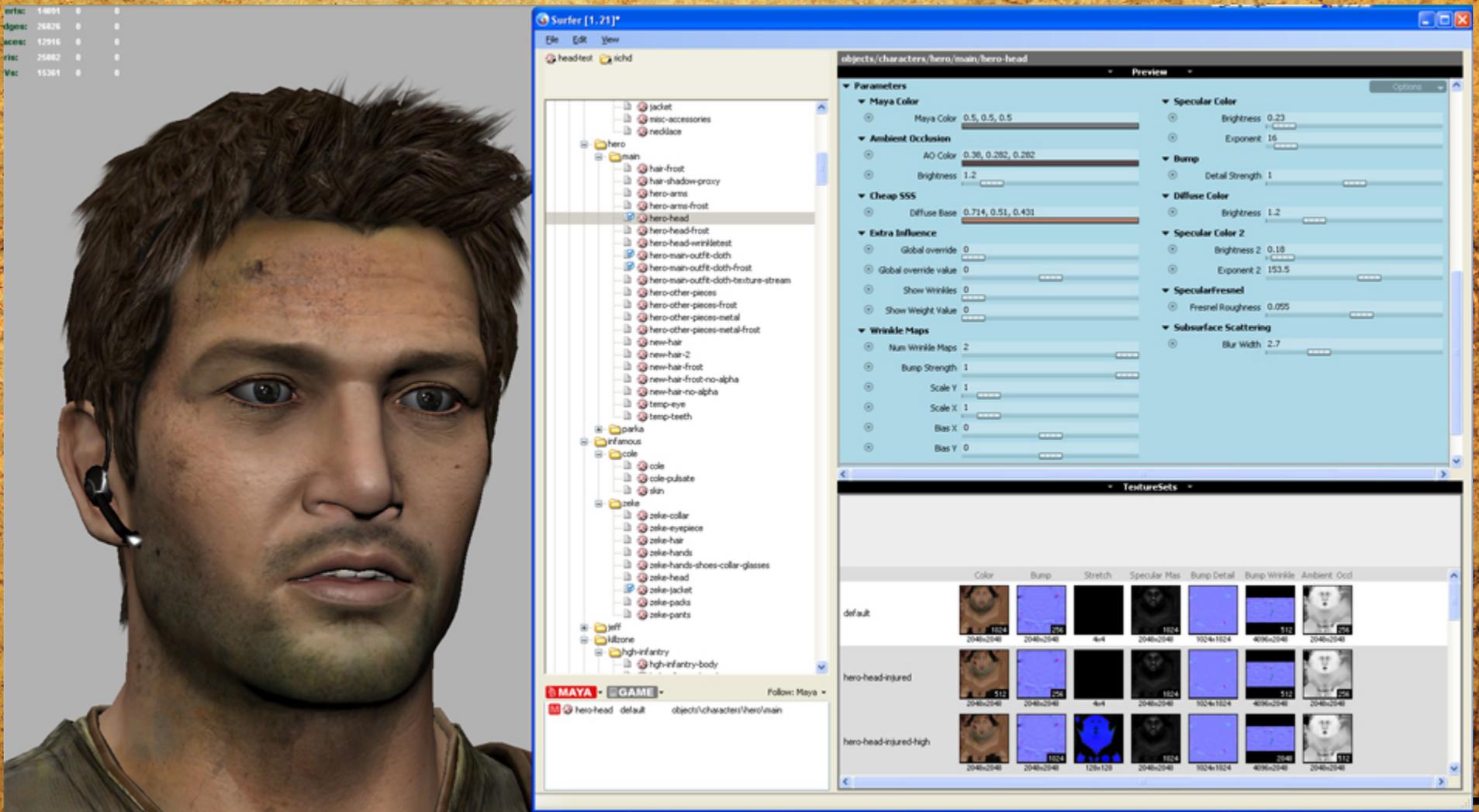
# Texturing/Sampling:

- Sample High resolution details from sculpt mesh to Arb game mesh (Normal Map)
- If high mesh was used to texture, sample color info, otherwise texture game mesh



# Shader Setup:

- Create shaders and assign the maps using our custom shader tools

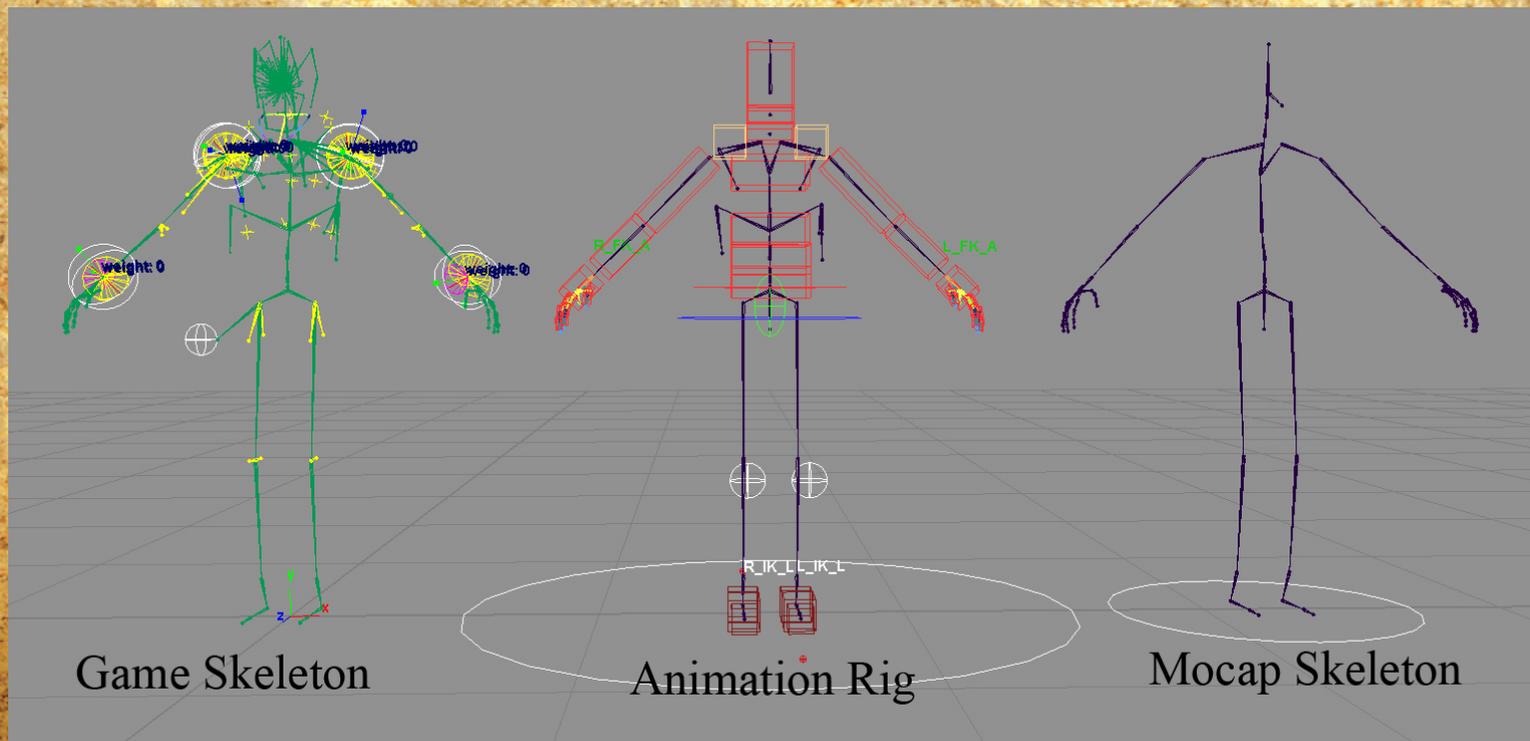


# Rigging Process:

- Rig Pipeline
- Deformation & Helper Joints
- Skinning Process
- Face Pipeline

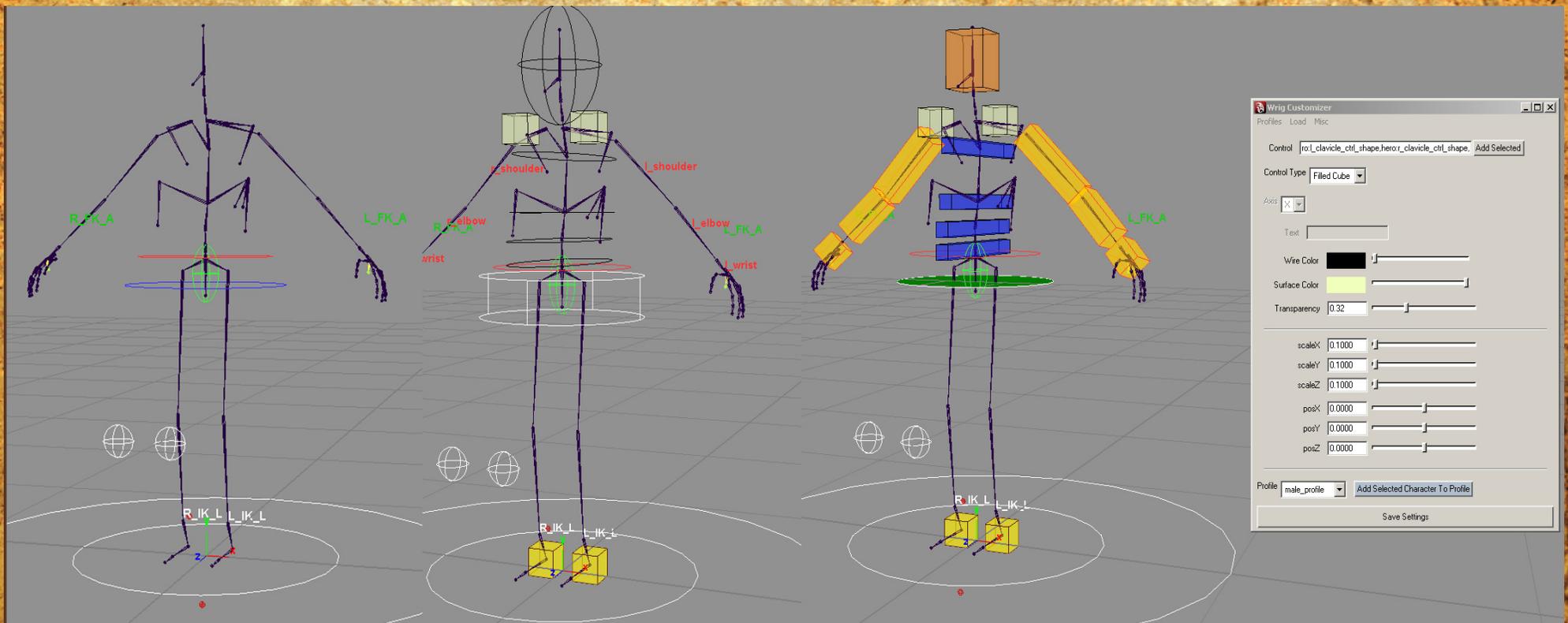
# Rig Pipeline:

- 3 separate skeletons:
- Skeleton Sharing: Males, Females, Children, Creatures
- All rigs have the same general orientation and naming conventions
- Main Character Joint #: 246



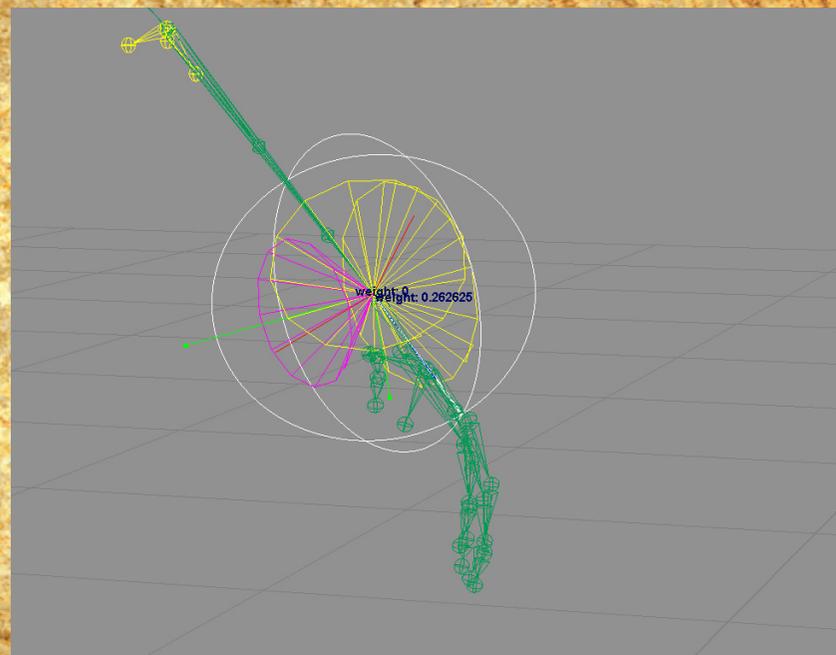
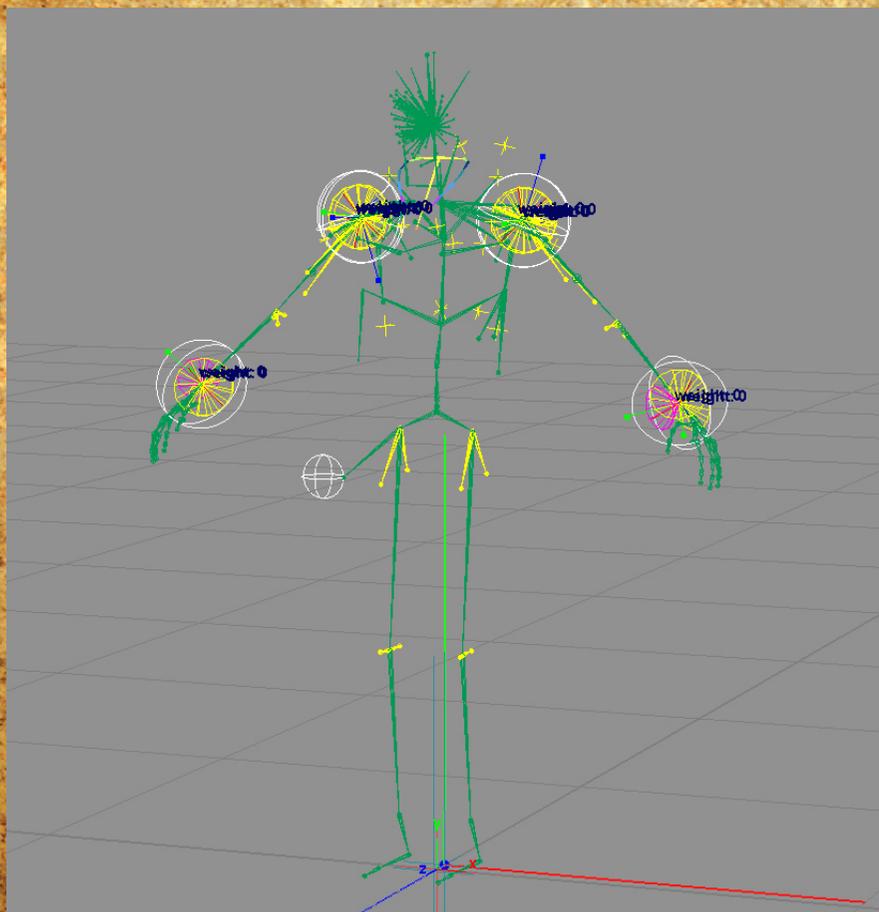
# Animation Rig:

- Standard Control Setup - kept simple to work with Motion Capture.
- Custom DG Node links between Motion Capture and Control Rig
- All Controls are custom OpenGL locators for customization



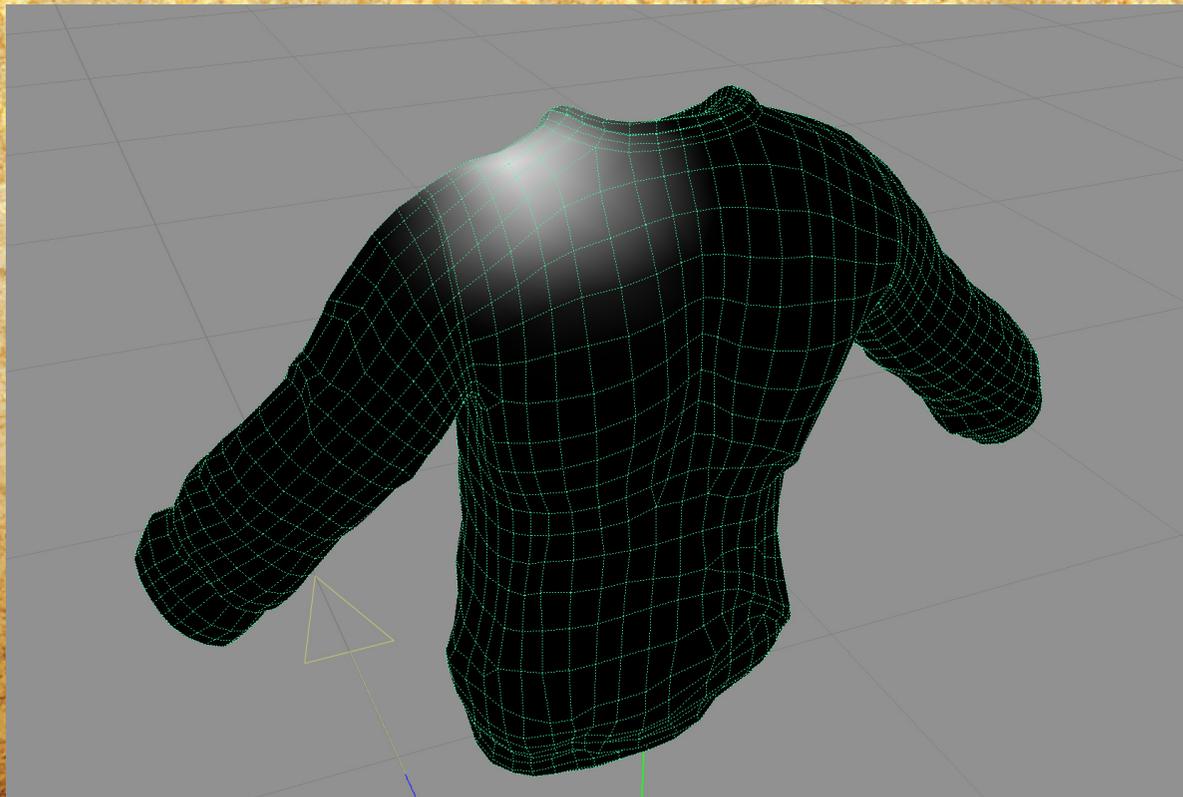
# Deformation & Helper Joints:

- All deformations are joints
- Helpers are run-time Set Driven Keys - no animation exported, saves memory. Limited use because of Gimbal.
- Use "Vector" cones to handle twists and more complex deformations.



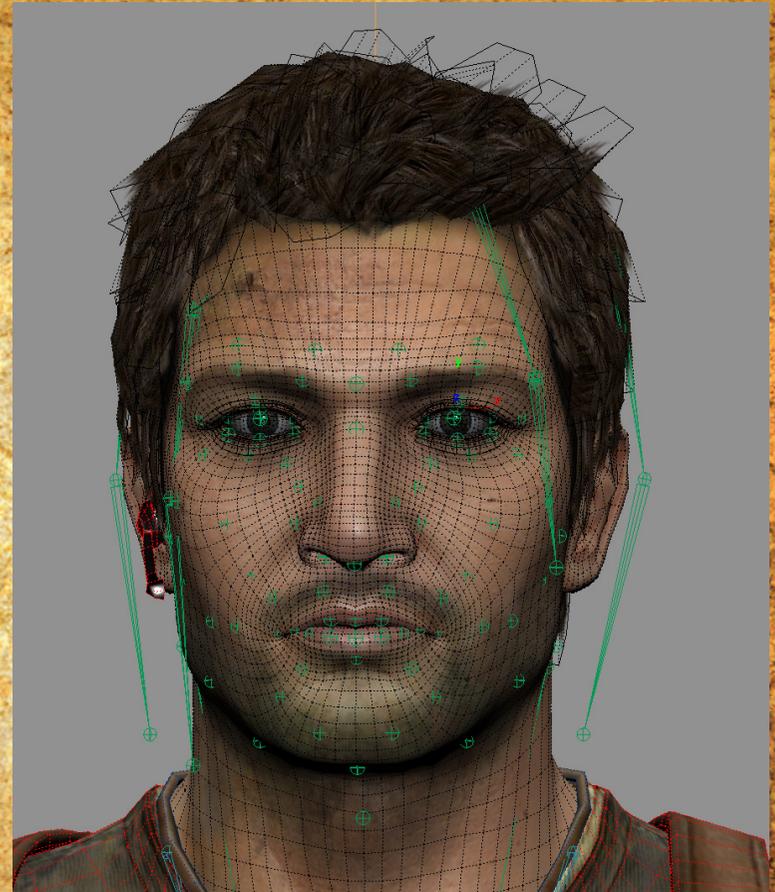
# Skinning Process:

- **Default Linear Blend Skinning**
- **Use Maya's "Copy Skin Weights" to get a basic start to new meshes**
- **History Tool allows modelers to modify topology - video**
- **3-5 influences per vertex - more than this becomes unmanageable.**
- **Use a motion capture range of motion to test the first pass, but the game itself is the best range of motion test.**

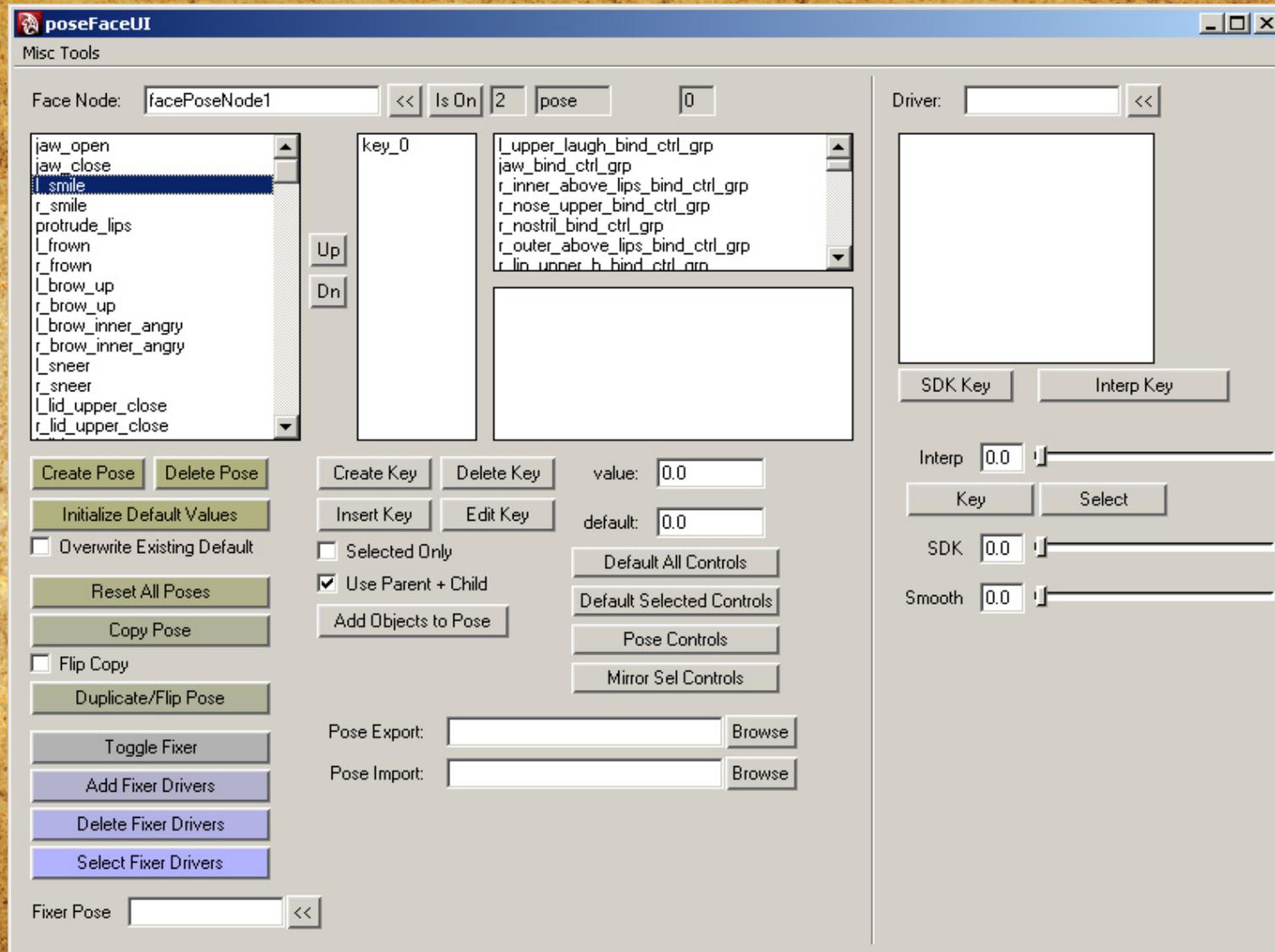


# Face Pipeline:

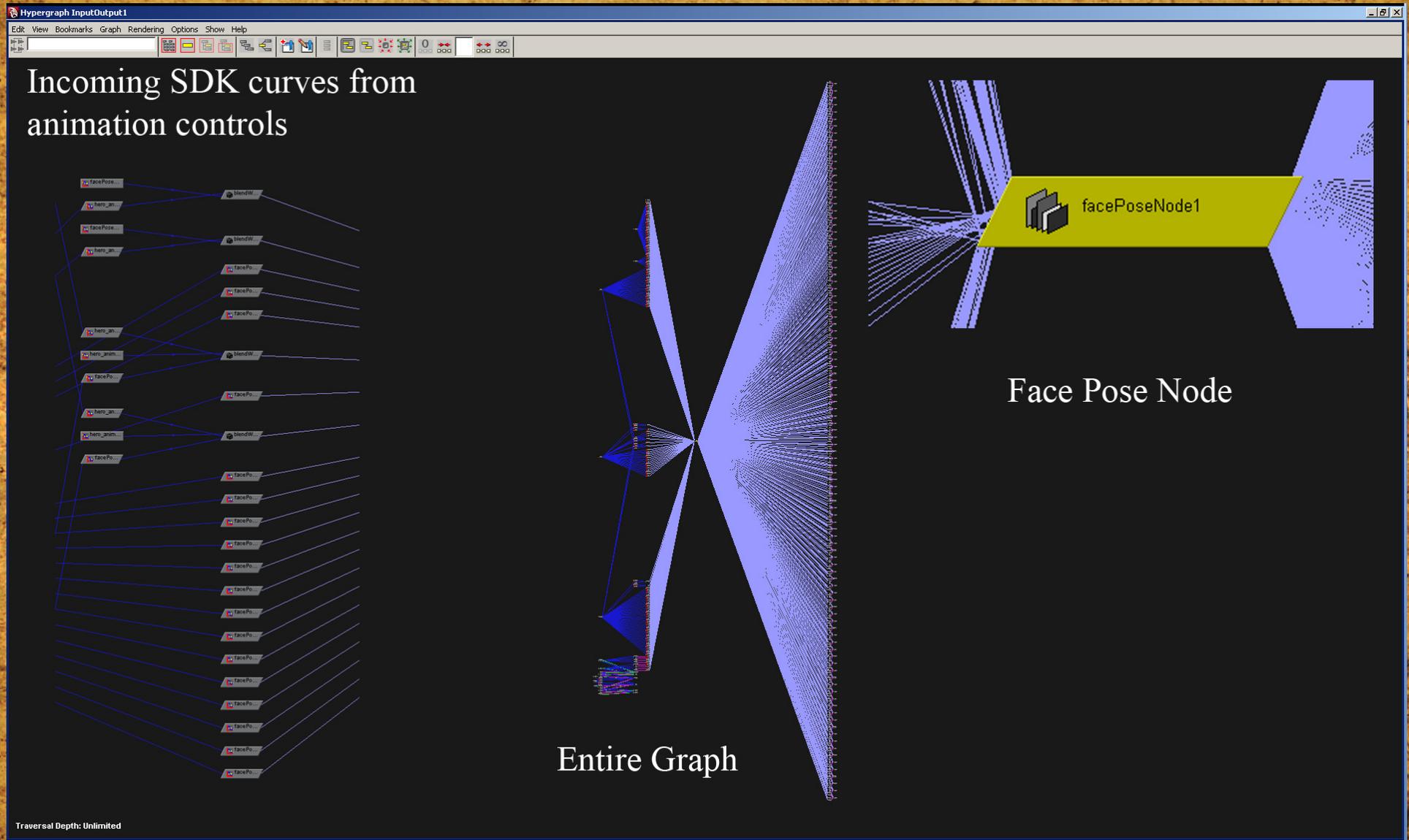
- Face Rig is all Joints
- 97 joints in the face
- Same facial rig In-Game & Cinematics
- Custom API node to hold all the pose data and do all the backend calculations
- UI that interfaces with the node and allows you to manage all the data



# Face Pipeline:



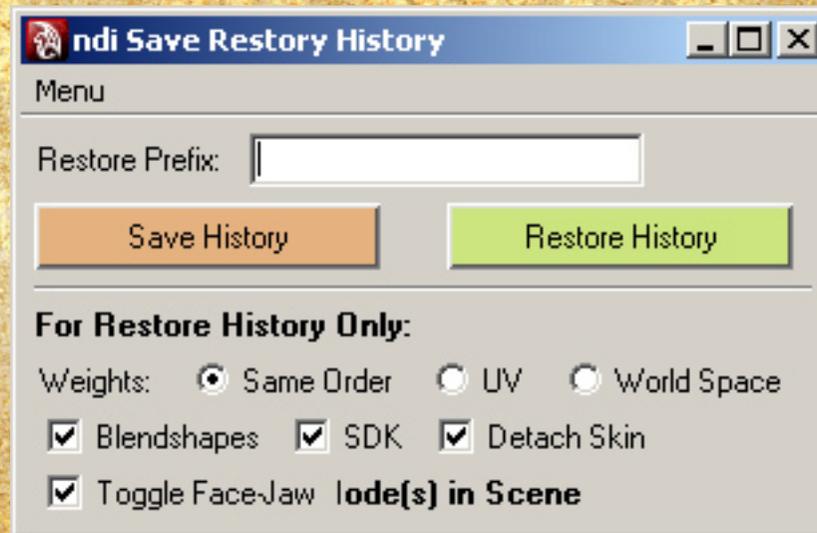
# Face Pipeline:



# Tools Overview:

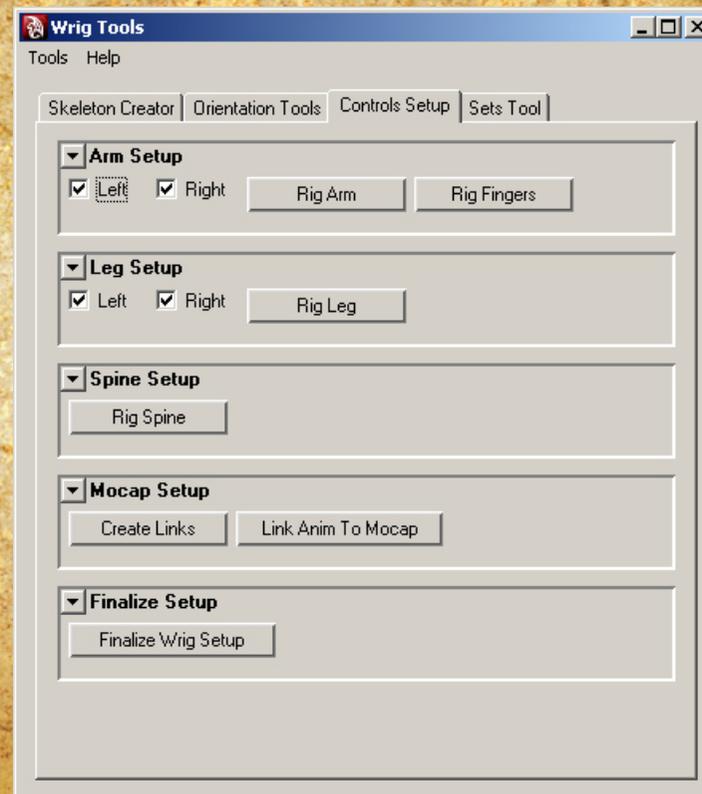
# Tools Overview:

## History Tool



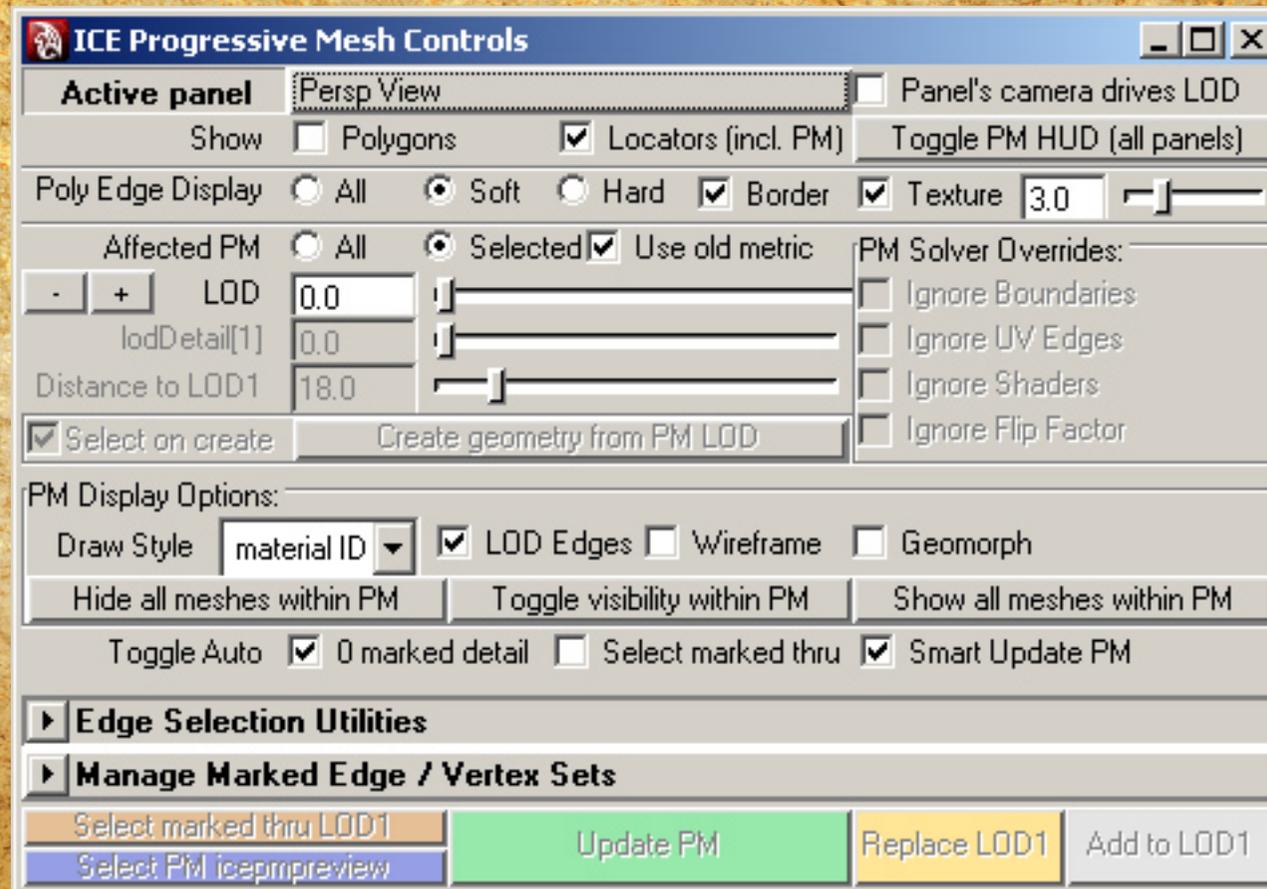
# Tools Overview:

## Auto Rig Builder



# Tools Overview:

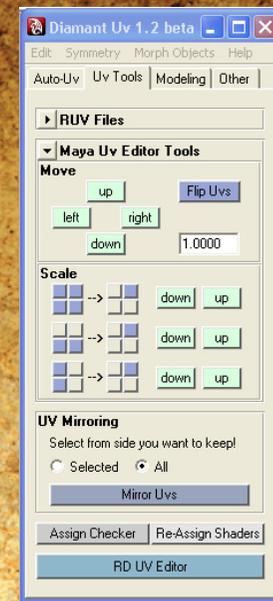
## PM LOD Tool





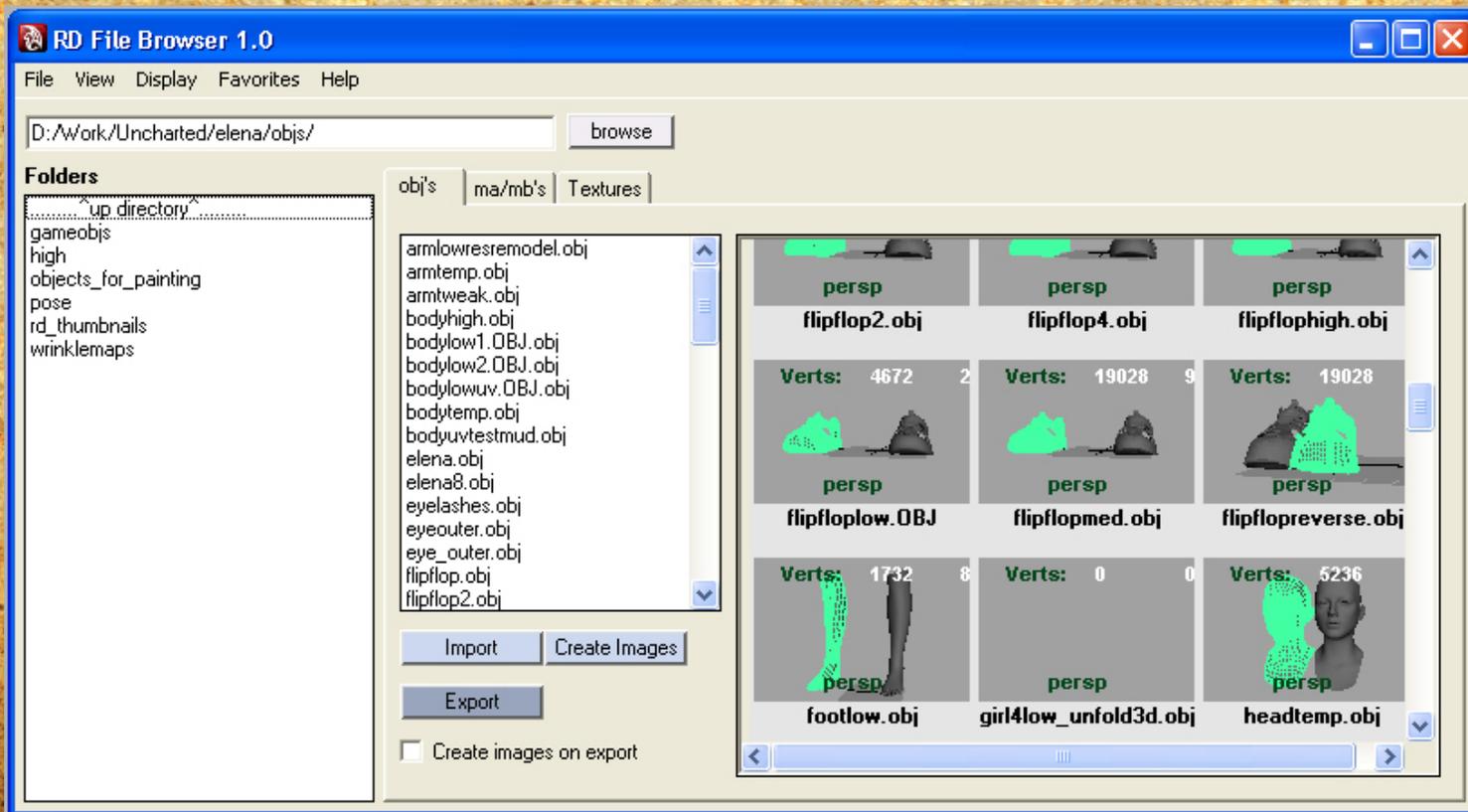
# Diamant UV:

- *Very fast uv creation and editing*
  - automatic uv generation
  - custom uv modifiers
- *Transferring single uv sets to models with multiple uv sets*
  - does not override both sets
- *Realtime shrink wrapping of in game models to high res models*
- *Topology transferring and uv transferring from one mesh to another that has the same topology but different point order*



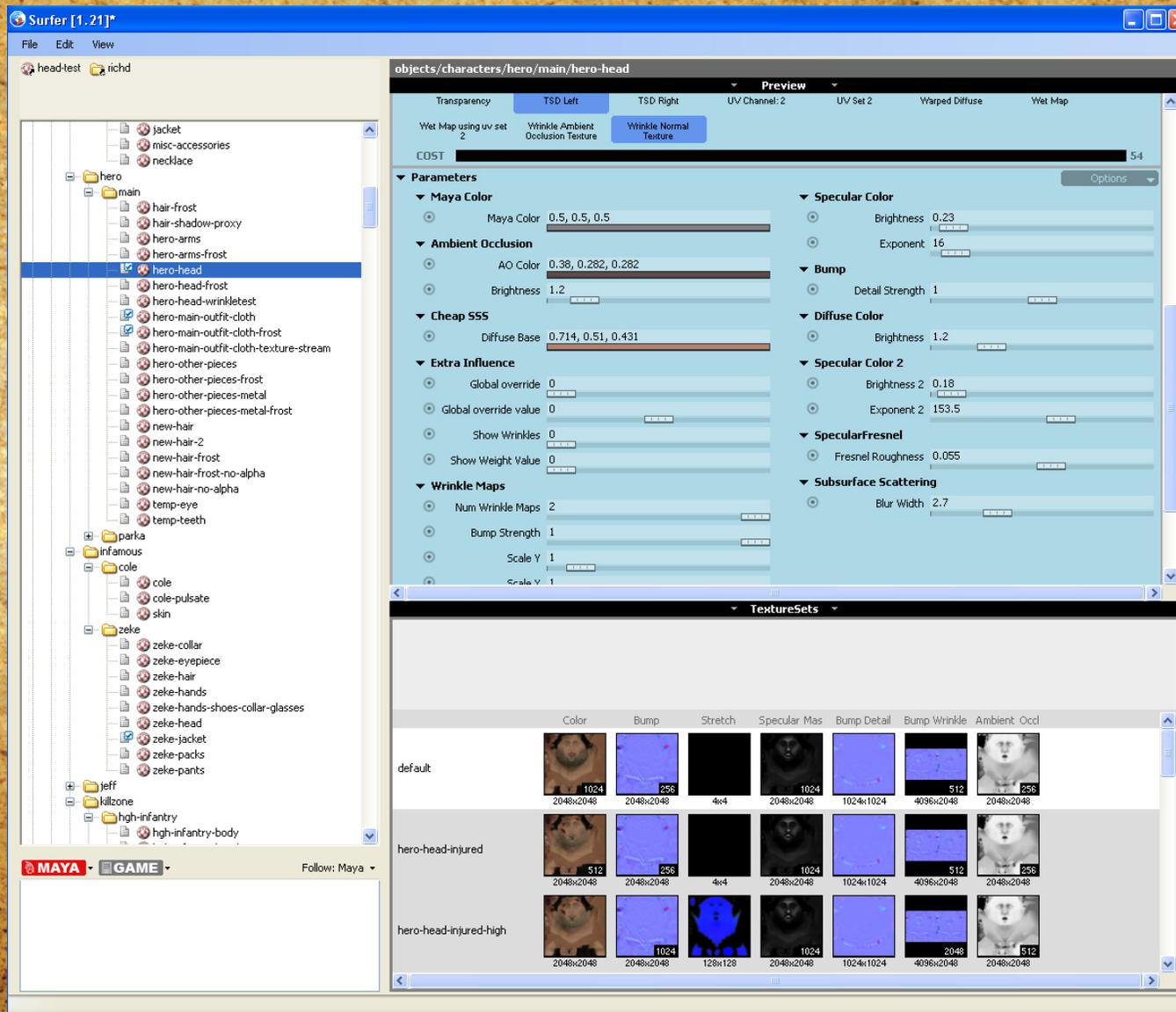
# Browser:

*Used to quickly browse and manage .obj files  
-quickly import and export multiple files at once*



# Tools Overview:

## Surfer



# What we wanted to fix from Uncharted 1

# Arbitrary Mesh Pipeline was too convoluted:

## The Problem:

*Uncharted 1 had two separate meshes for every part.*

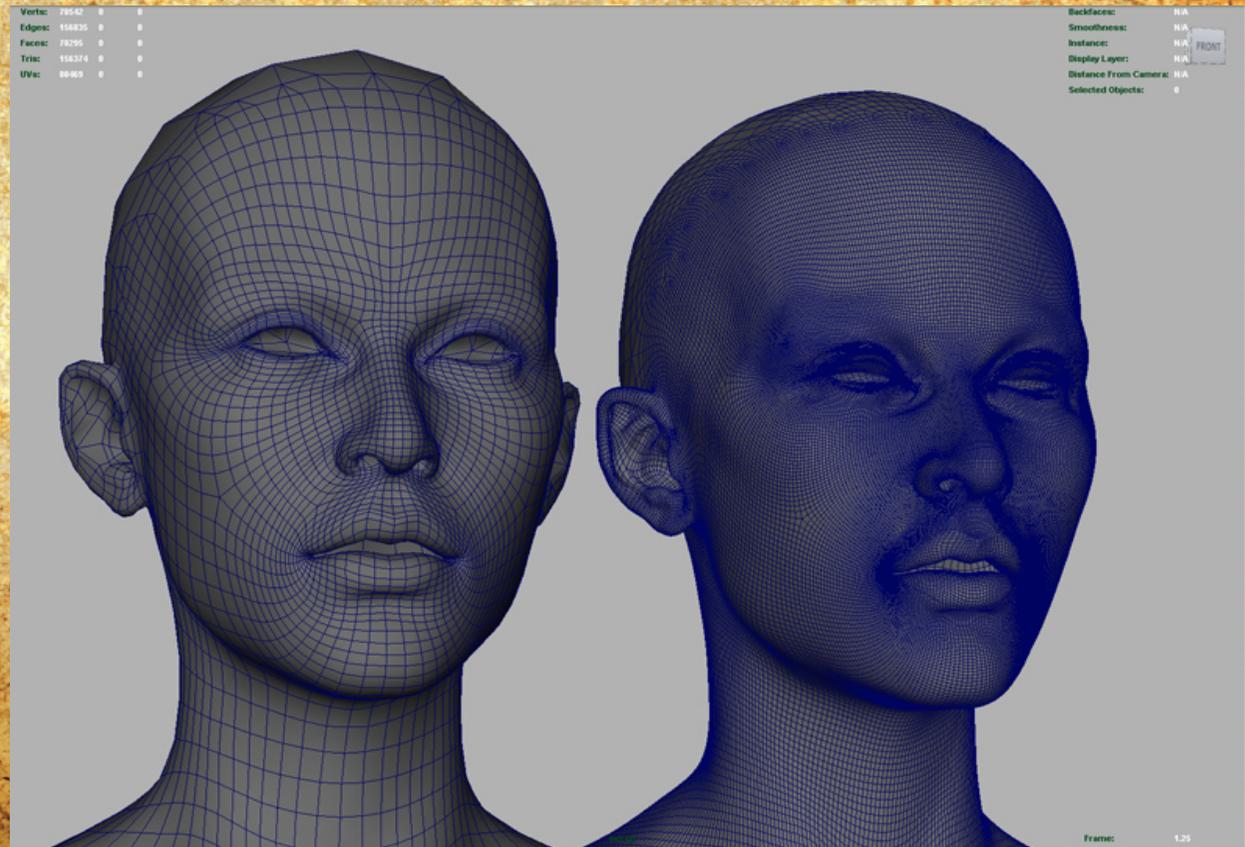
*Problems when updating either of the two meshes which caused inconsistencies and management issues.*

*Major issue when dealing with the creation of wrinkle maps since the poses were created on the game mesh*

# Arbitrary Mesh Pipeline was too convoluted:

## Partial Solution:

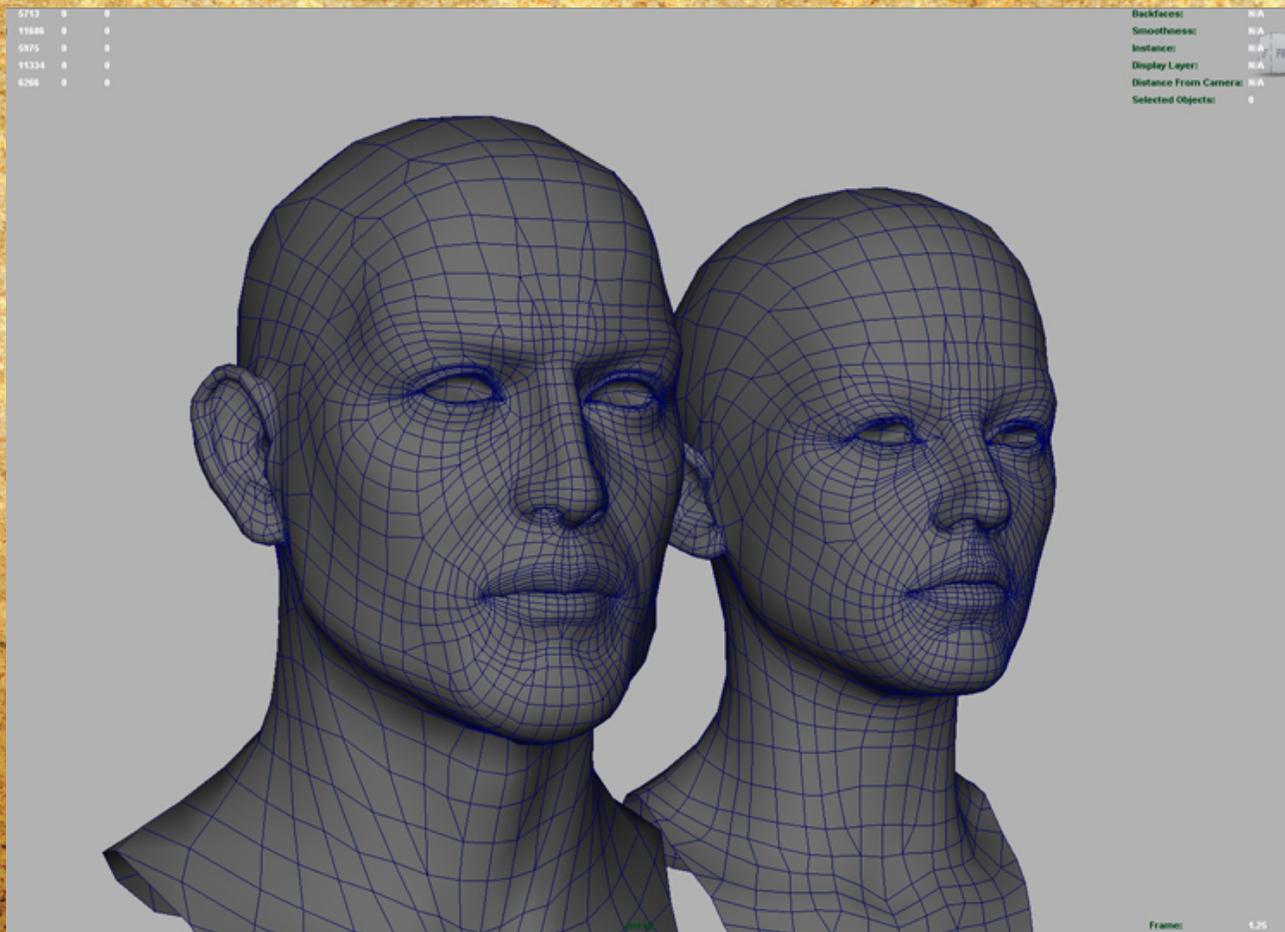
- *Used the same base head for the high res sculpt and the game mesh.*



# Different topology for each of the characters heads:

## Problem:

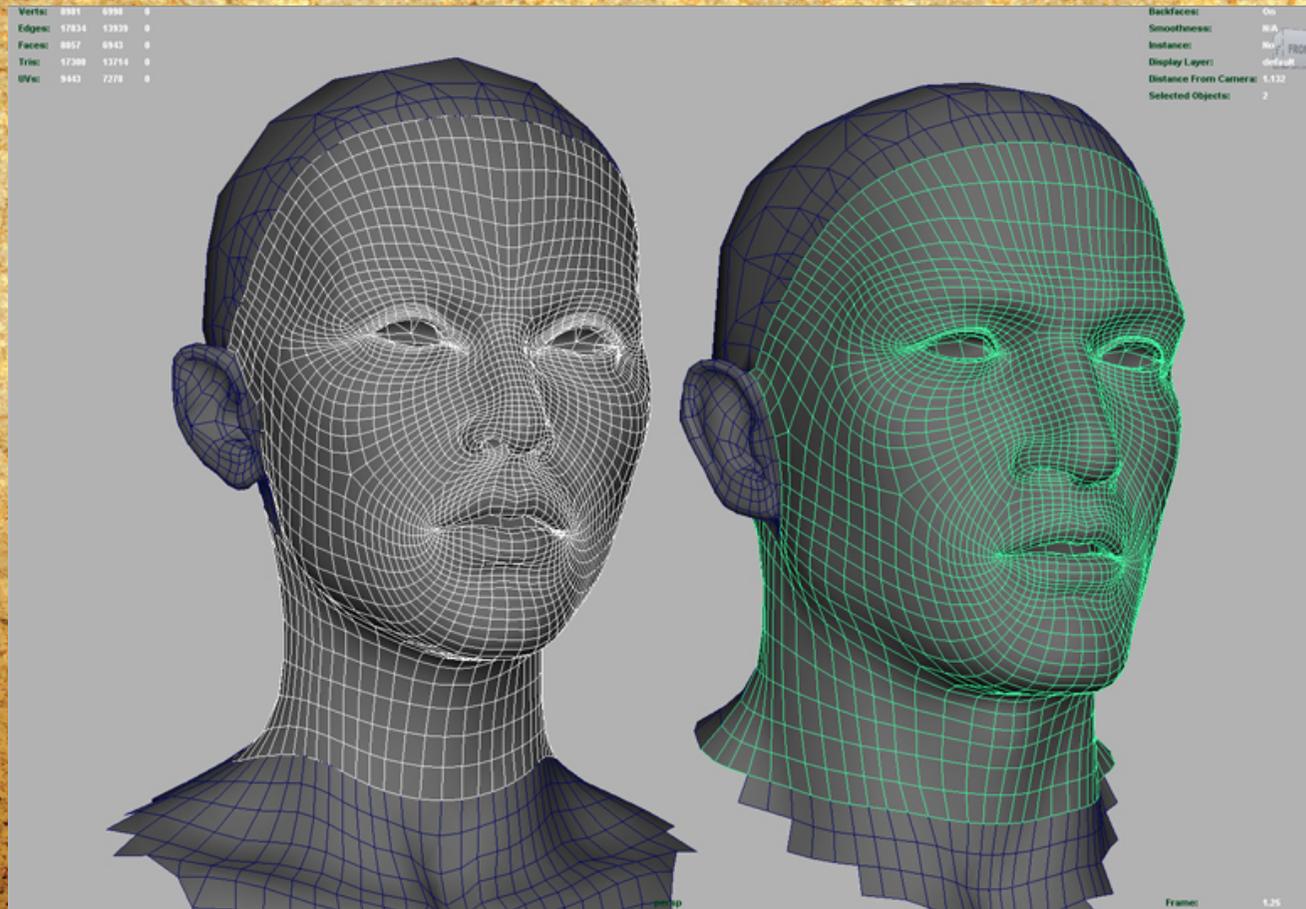
- *Each character had different topology*
  - *each character had to be rigged from scratch*
  - *creating the arbitrary game meshes took a long time*



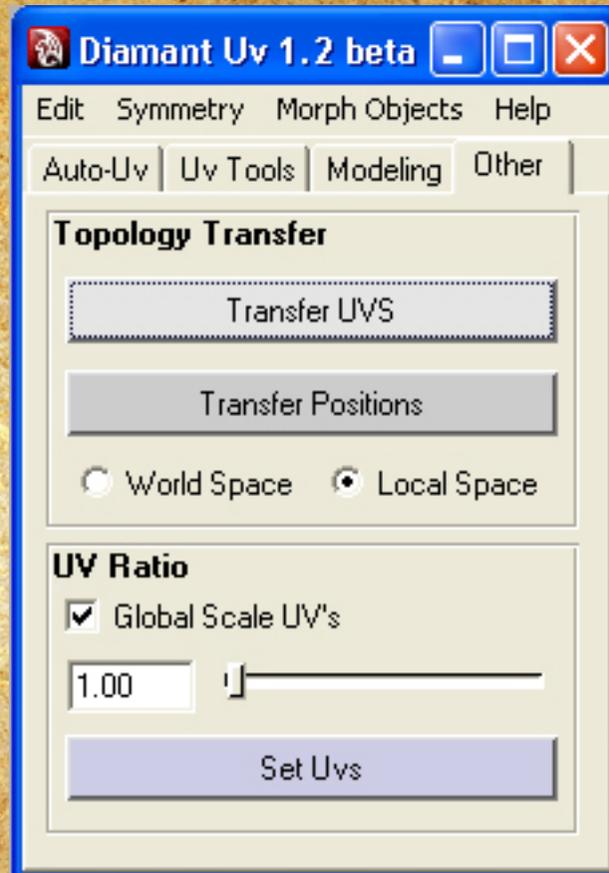
# Different topology for each of the characters heads:

## Solution:

- *Created a standard Head mesh that every character used. (base section)*
- *made creating the game mesh extremely fast*
- *allowed us to transfer weighting to all main characters*



# Video: Transferring process using Diamant UV tool



# Re-topologizing game mesh :Video

Verts: 8831 8831 0  
Edges: 17809 17809 0  
Faces: 8980 8980 0  
Tris: 17548 17548 0  
UVs: 13742 13742 0

Backfaces: N/A  
Smoothness: N/A  
Instance: No FRONT  
Display Layer: default  
Distance From Camera: 0.888  
Selected Objects: 2



# Rigging Problems:

JS

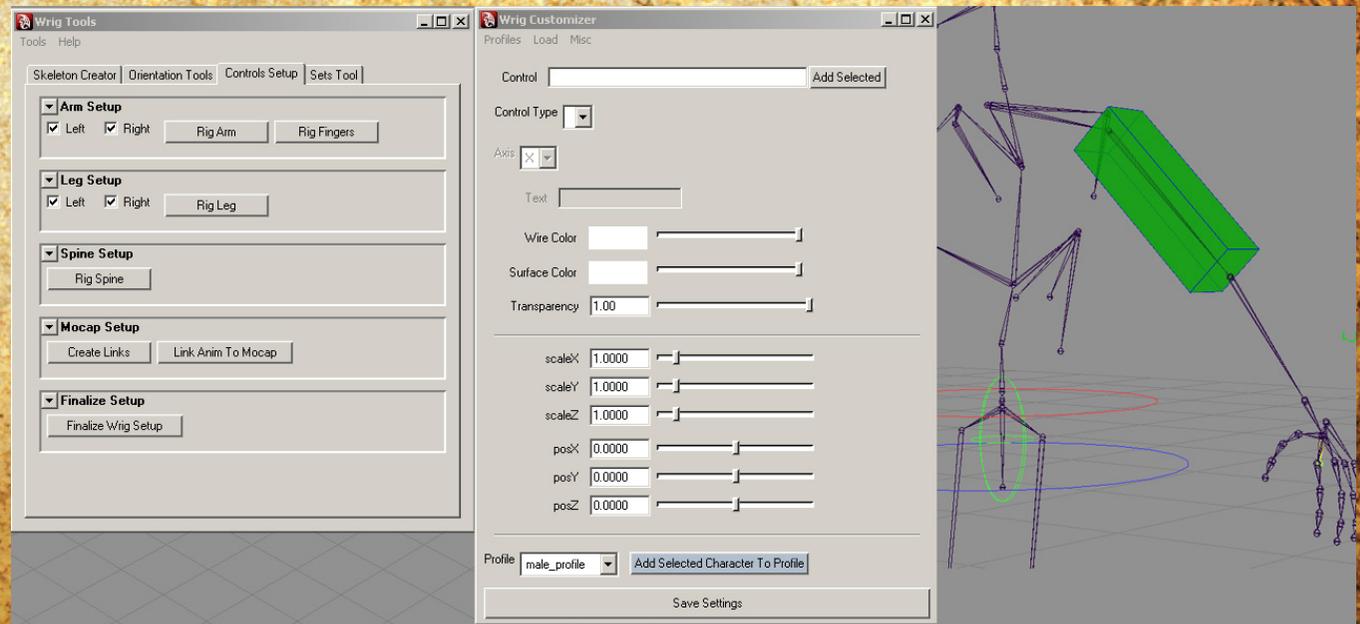
# Building a rig from scratch was a nightmare:

## The Problems:

- We had no easy way of automating the setup.
- No way of keeping consistent info across all the joints and the control setups (orientation of controls and joint orientation).
- Animators can't transfer animation across skeletons.

# Solution:

- Generalized the rig setup to allow for automation.
- Wrote orientation procedure that would calculate consistent orientation across all skeletons.
- Created general rig control system to simply curve control creation.



# Rig Builder and Animation on different Skeletons:

*Video:*



# Character Faces:



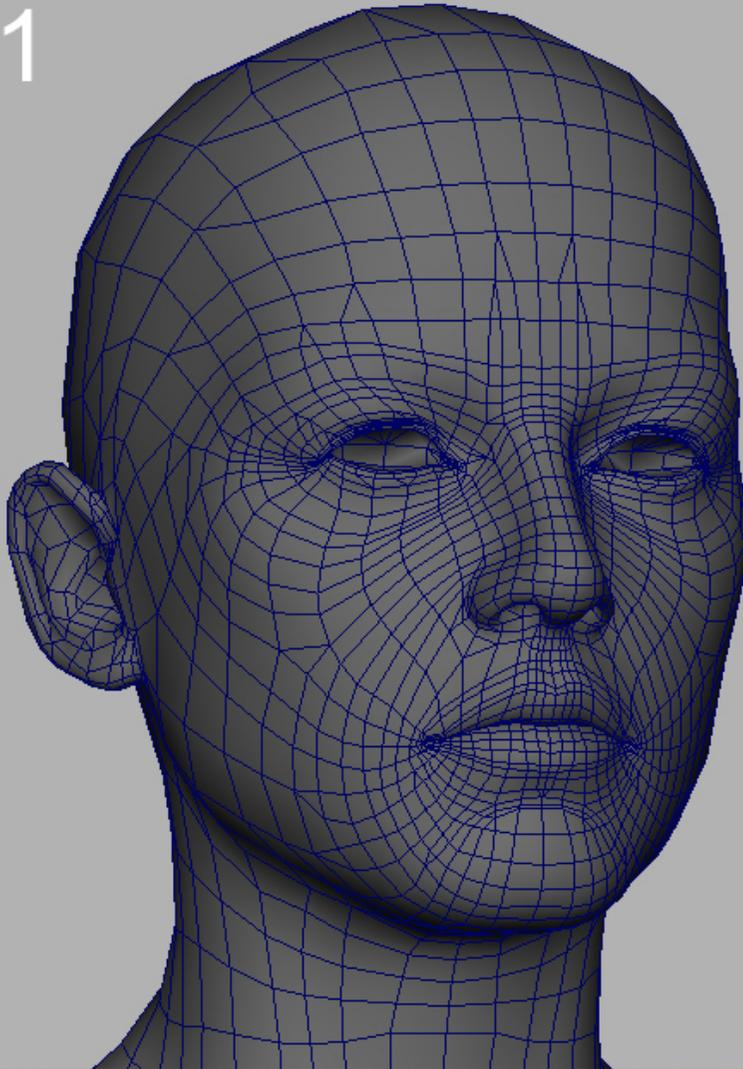
RD

# Geometry in the faces and expressions:

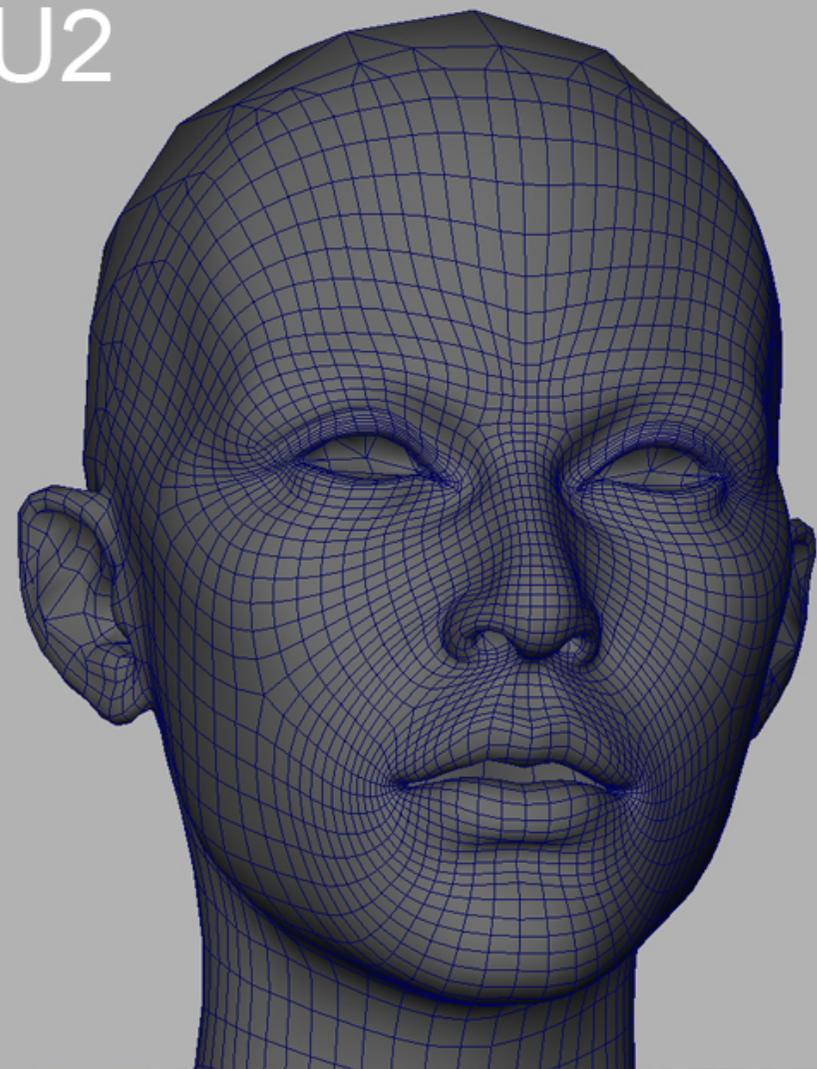
- Wanted to achieve a higher level of fidelity with the faces and expressions.
- Needed to add more geometry and new topology
- New facial rig with significantly more bones

# Geometry in the faces and expressions:

U1



U2



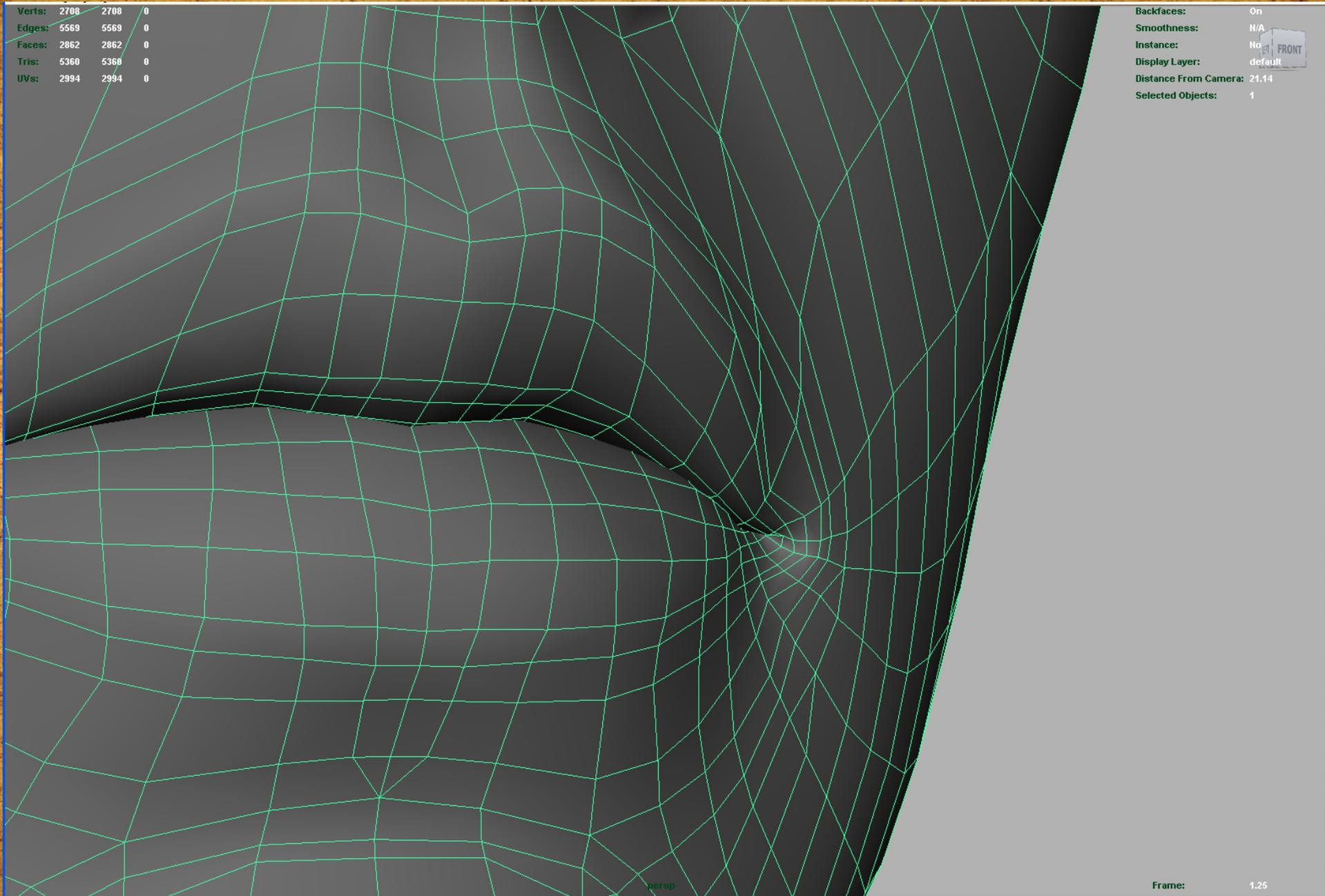
# Mouth area:

- Uncharted 1's mouth area was very hard to weight and didn't look very good.
- Changed to an open mouth start pose
- Added more geometry that flowed better for deformation
- Used custom tools to smooth the noise between verts

# Mouth area: Uncharted 1

Verts:	2708	2708	0
Edges:	5569	5569	0
Faces:	2862	2862	0
Tris:	5360	5360	0
UVs:	2994	2994	0

Backfaces:	On
Smoothness:	N/A
Instance:	No 
Display Layer:	default
Distance From Camera:	21.14
Selected Objects:	1



persp

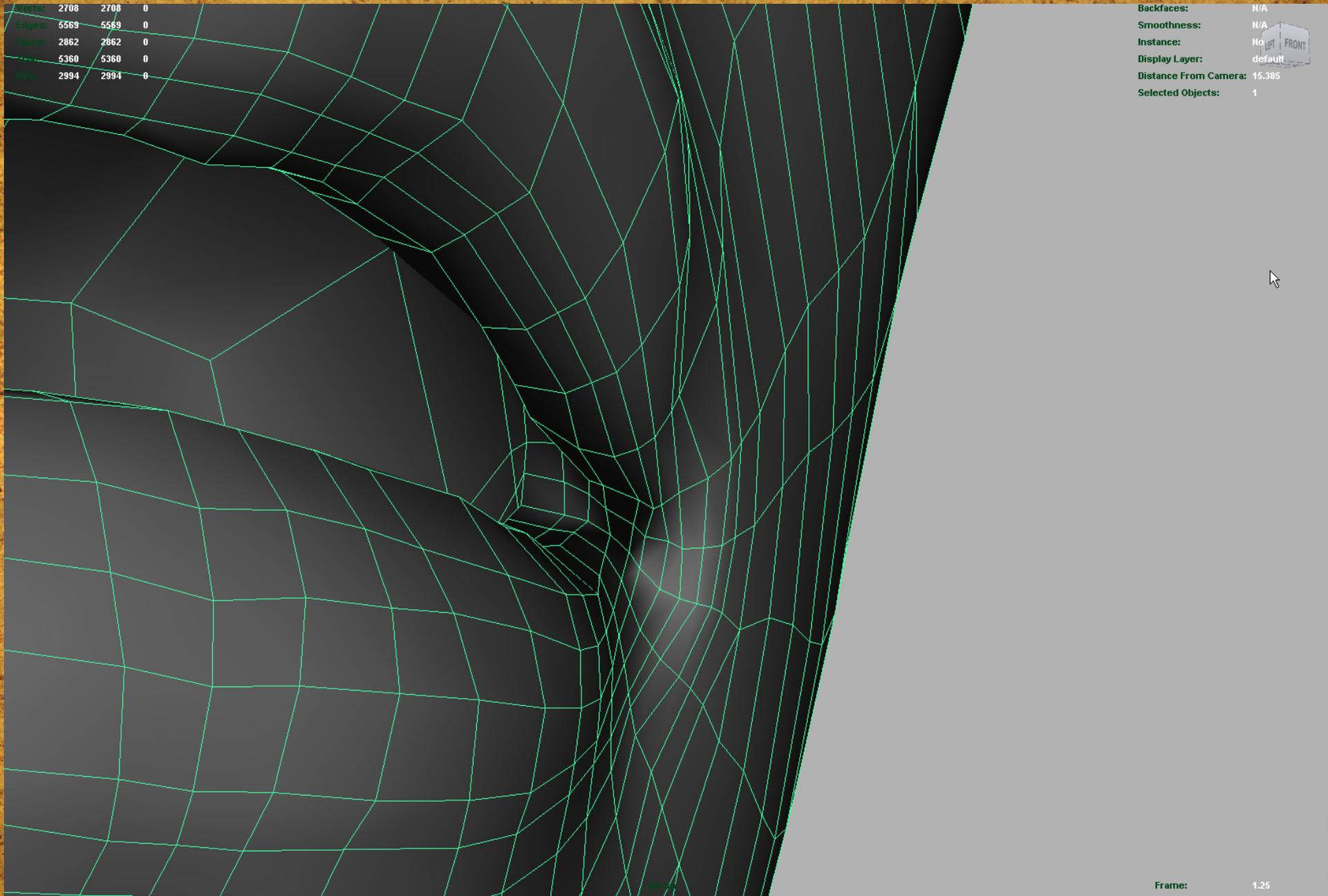
Frame: 1.25



# Mouth area: Uncharted 1

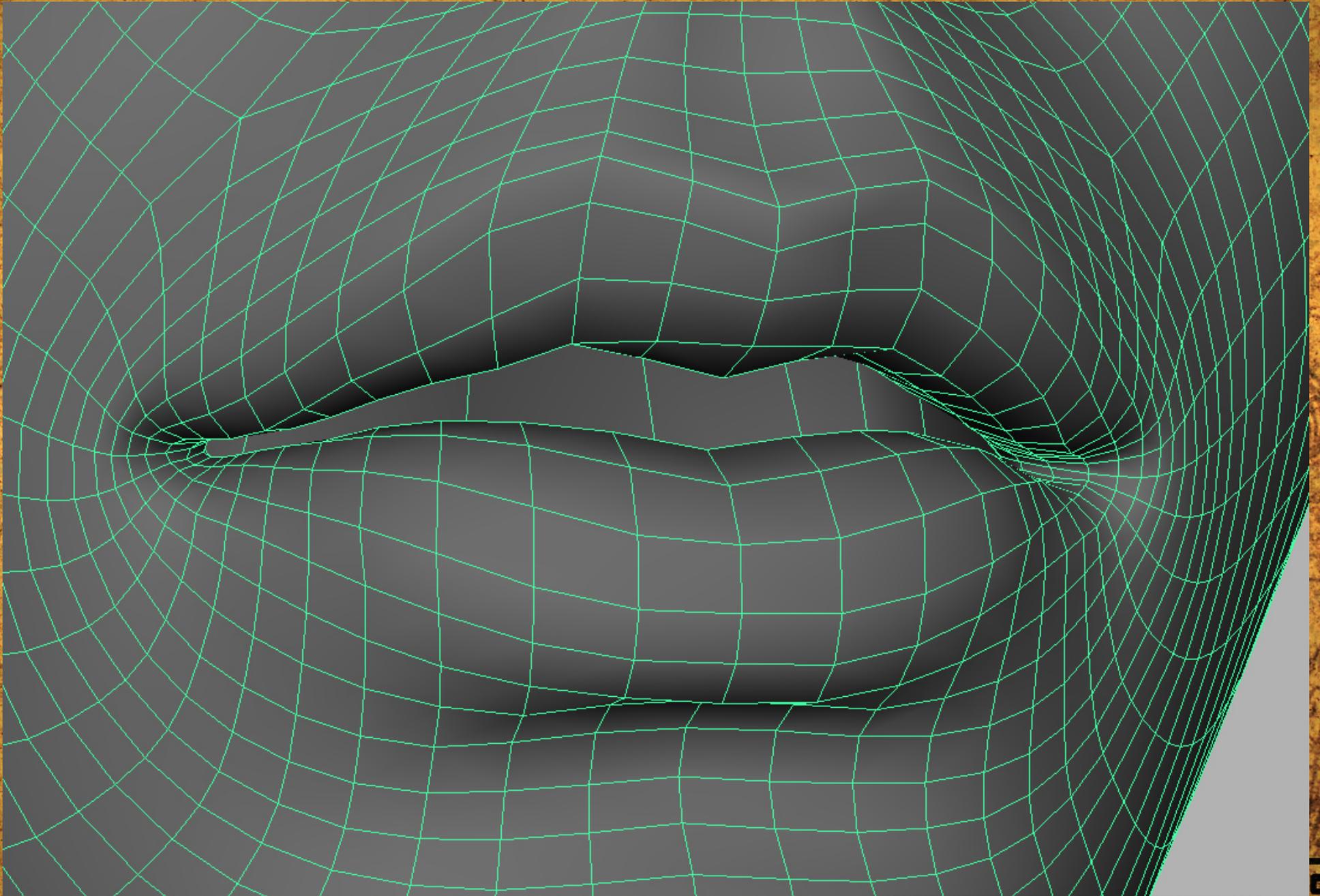
Verts: 2708 2708 0  
Edges: 5569 5569 0  
Faces: 2862 2862 0  
UVs: 5360 5360 0  
Bones: 2994 2994 0

Backfaces: N/A  
Smoothness: N/A  
Instance: No  
Display Layer: default  
Distance From Camera: 15.385  
Selected Objects: 1

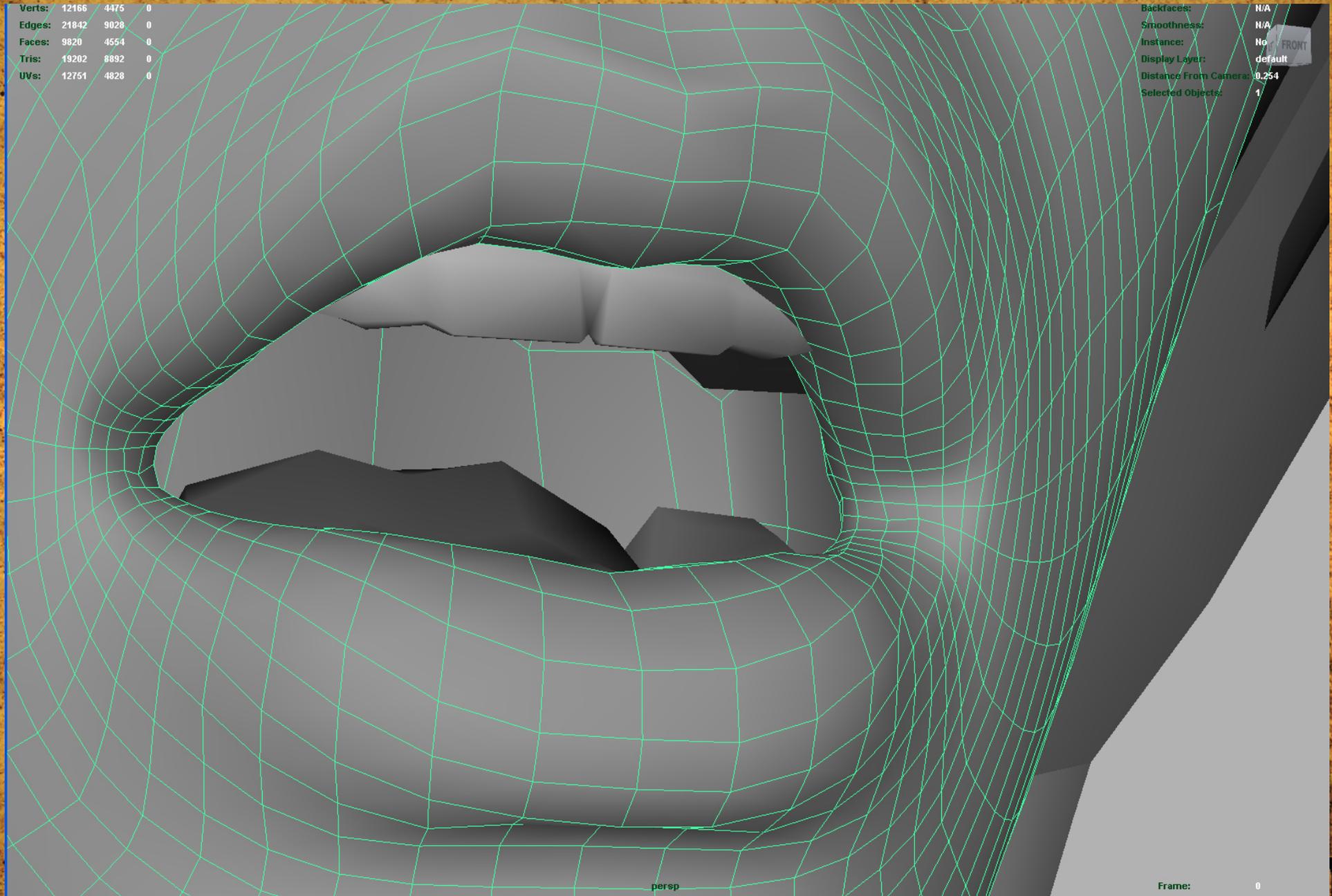


Frame: 1.25

# Mouth area: Uncharted 2



# Mouth area: Uncharted 2



# Facial Controls:

## Problem:

- Uncharted 1 had Viewport Controls
  - Animators couldn't easily see what was on and what the values are
  - Selection is annoying
  - Doesn't scale up well

## Solution:

- Switched to Channel Box sliders.



mouth_ctrl	
Mouth_Overall	-----
Jaw_open	0
Jaw_open_relaxed_corners	0
Jaw_jut	0
Jaw_left_right	0
Mouth_back_forward	0
Mouth_left_right	0
Mouth_left_right_no_twist	0
Mouth_up_down	0
Close_lips	0
Tighten_lips	0
Mouth_Emotion	-----
L_smile	0
R_smile	0
L_frown	0
R_frown	0
Pucker_Shapes	-----
Pucker	0
Potrude_pucker	0
Tighten_protrude	0
Ssh_mouth	0

# Facial Rig/Shapes:

- Spent some time researching more anatomy
- Try to maintain bone structure and give the feeling of skin and muscle moving over bone
- Better understanding of how the face works
- Give more control to the animators

# Face Comparison:



# Face Comparison:



# Face Comparison:



# Fixer Shapes:

- Use more fixers to make sure when poses come together we get more anatomically correct shapes and more appealing shapes - Video



# Improve the eyes:

*Uncharted 1 Eye: Not Grounded*



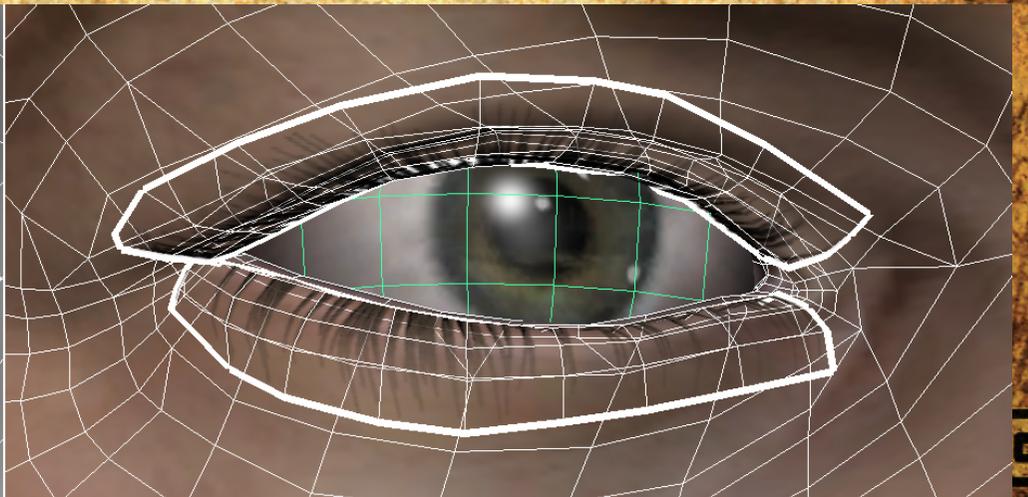
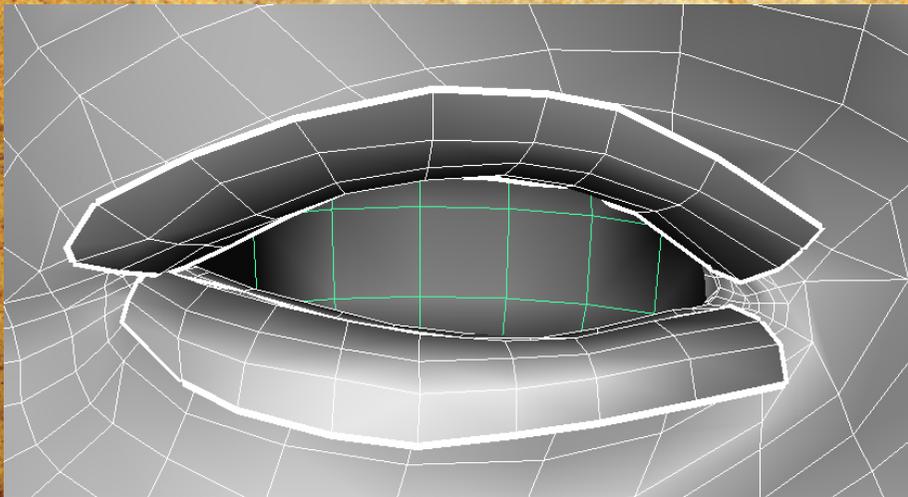
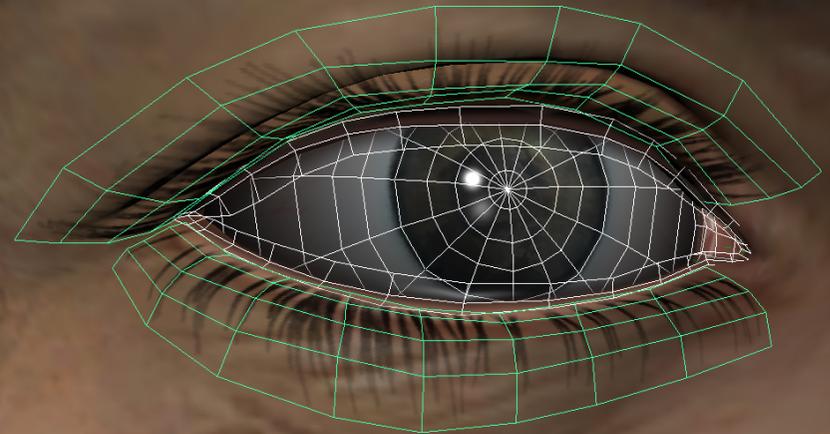
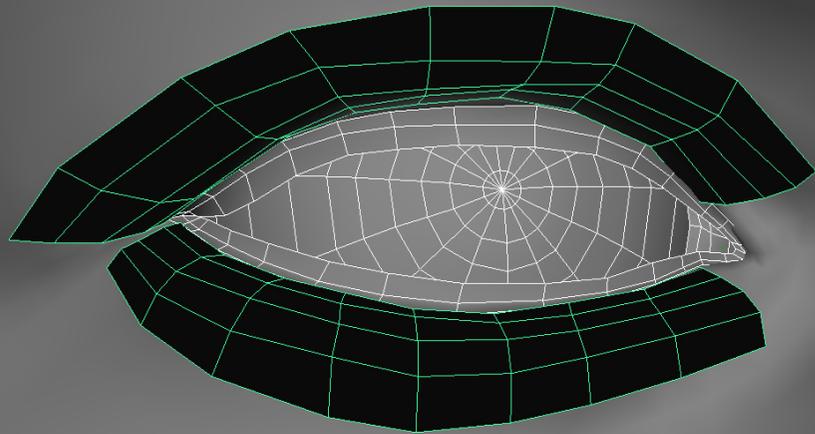
RD

# Uncharted 2 Eyes:



# Improve the eyes:

*Uncharted 2 Eyes*



# Eye Compare:

Uncharted 1

Uncharted 2



# Quick iteration times between sculpting and in game:

*Experimented with getting the quaded Zbrush mesh skinned quickly and in game for approval before building the game mesh.*

## *Pros:*

*-we didn't go too far into the character without seeing it in game first*

## *Cons:*

*-Often didn't keep track of this and had to reskin things too many times*

# Quick iteration times between sculpting and in game:



# Quick iteration times between sculpting and in game:



# New challenges specific to Uncharted 2

# New challenges specific to Uncharted 2:

- LODs
- Multiplayer skins
- Optimization
- Outsourcing
- Villagers last minute
- Normals not transforming based on translation for the eyes
- Improved Hair

# LODs:

- Had not used lods in Uncharted 1.
- Had to keep vertex sets low for Uncharted 2.
- Decided to do this late in production.



# LODs:

- First started manually creating Lods
- Used Maya PM Lod Tool to generate Lods.
- Used Maya "Copy Skin Weights" from base geometry to all Lods. - Video



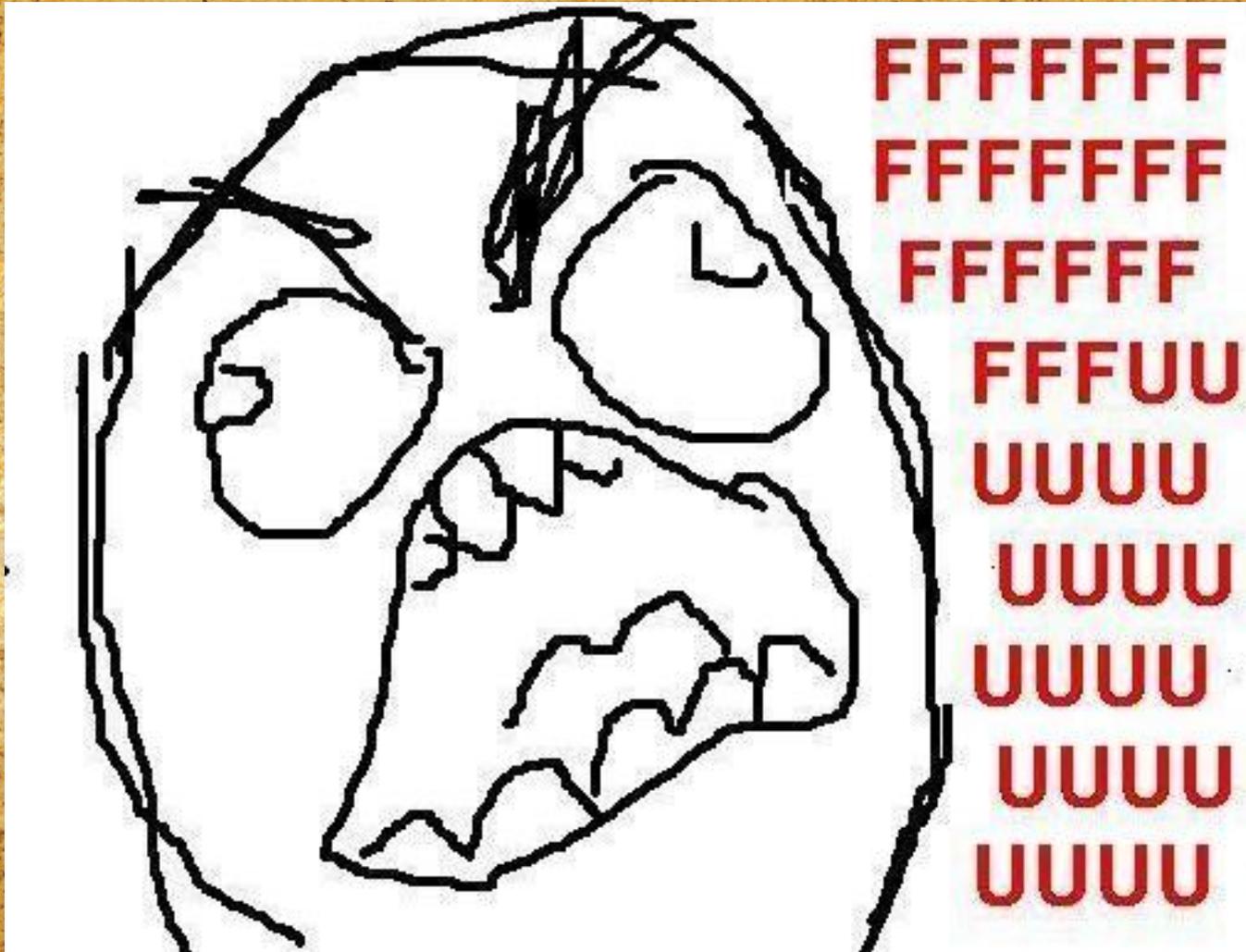
# The Completely Unexpected

# Multiplayer



st

# Multiplayer

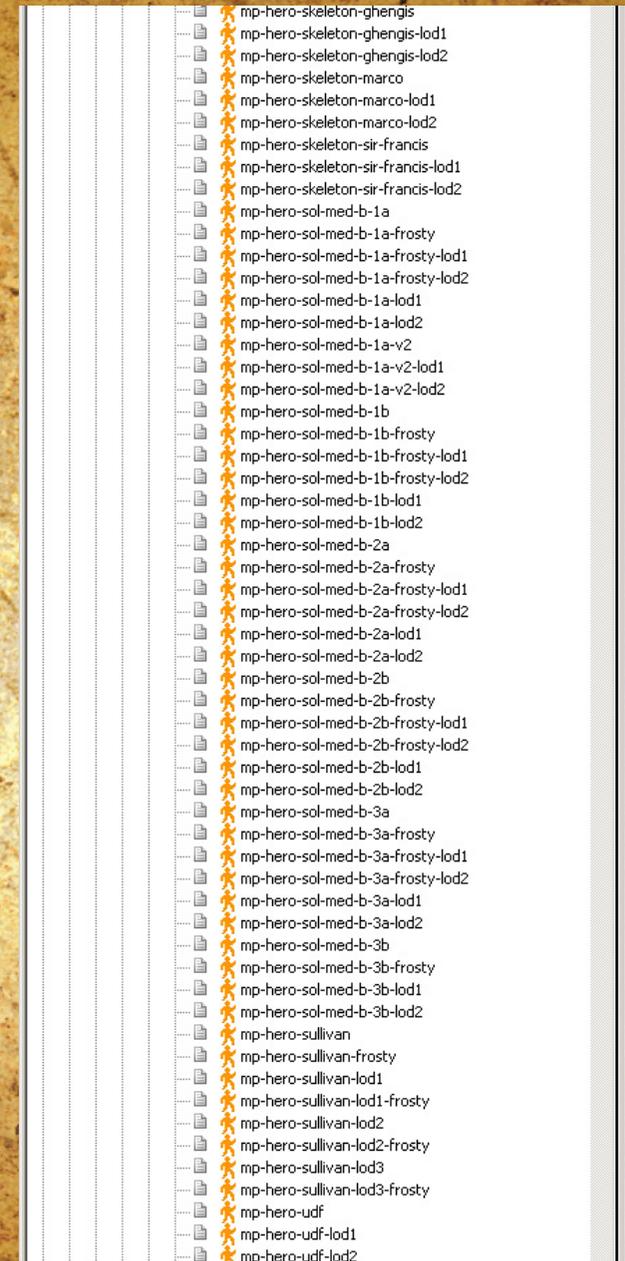


# Multiplayer

- Needed Lods for each Multiplayer skin.
- Had tons of Build Assets to manage
  - Separate actor for each MP skin and LODs
- One skeleton for all characters
- Needed facial animation shared

# Multiplayer

- Roughly 50 MP skins generated
- Had tons of Build Assets to manage
  - Separate actor for each MP skin and LODs



# Multiplayer

- One skeleton for all characters
  - Females proportions were really hard to handle
- Needed facial animation shared - video



# Optimization:

- Did not anticipate having to optimize as much
  - -weren't as tight on Uncharted 1.
- Vertex sets needed to be kept low.
  - used Lods
  - combined geometry into single pieces
  - used less shaders
  - combined textures to fit on single texture sheets
- Turned off shadow casting on smaller objects
- Turned off motion blur and other shader parameters on smaller objects

# Villagers last minute:

*Initially only meant to be used in the far background and not shown up close.*

*Very last minute change to add facial expressions and up the quality for a better look.*



# Villagers last minute:



RD



# Normals not transforming based on translation for the eyes:

*Eye lids are driven by multiple joints right at the surface of the lid.*

*-Translation is used to transform the vertices.*

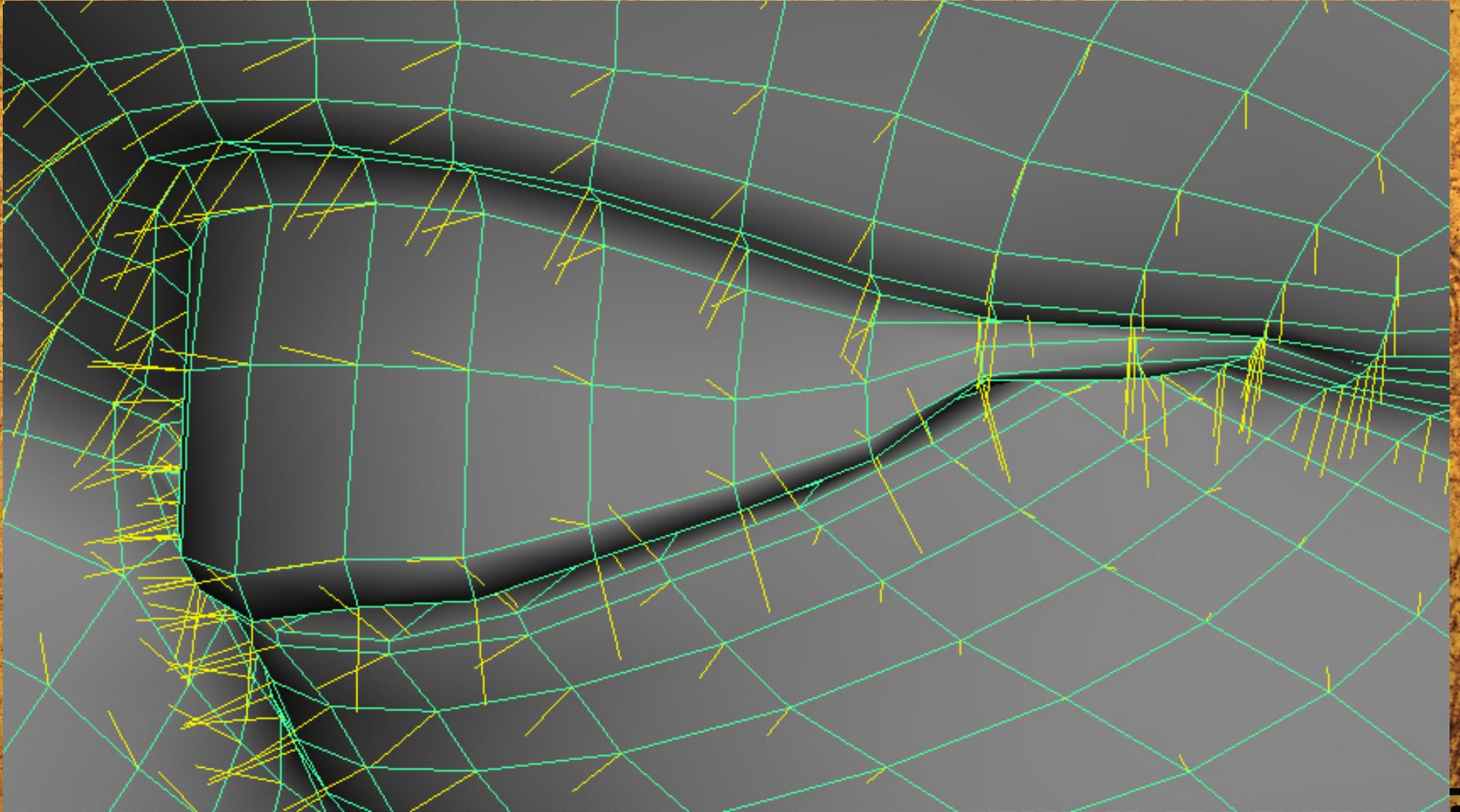
*Causes artifacts due to normals not being transformed correctly.*

*Unavoidable due to technology limitations.*

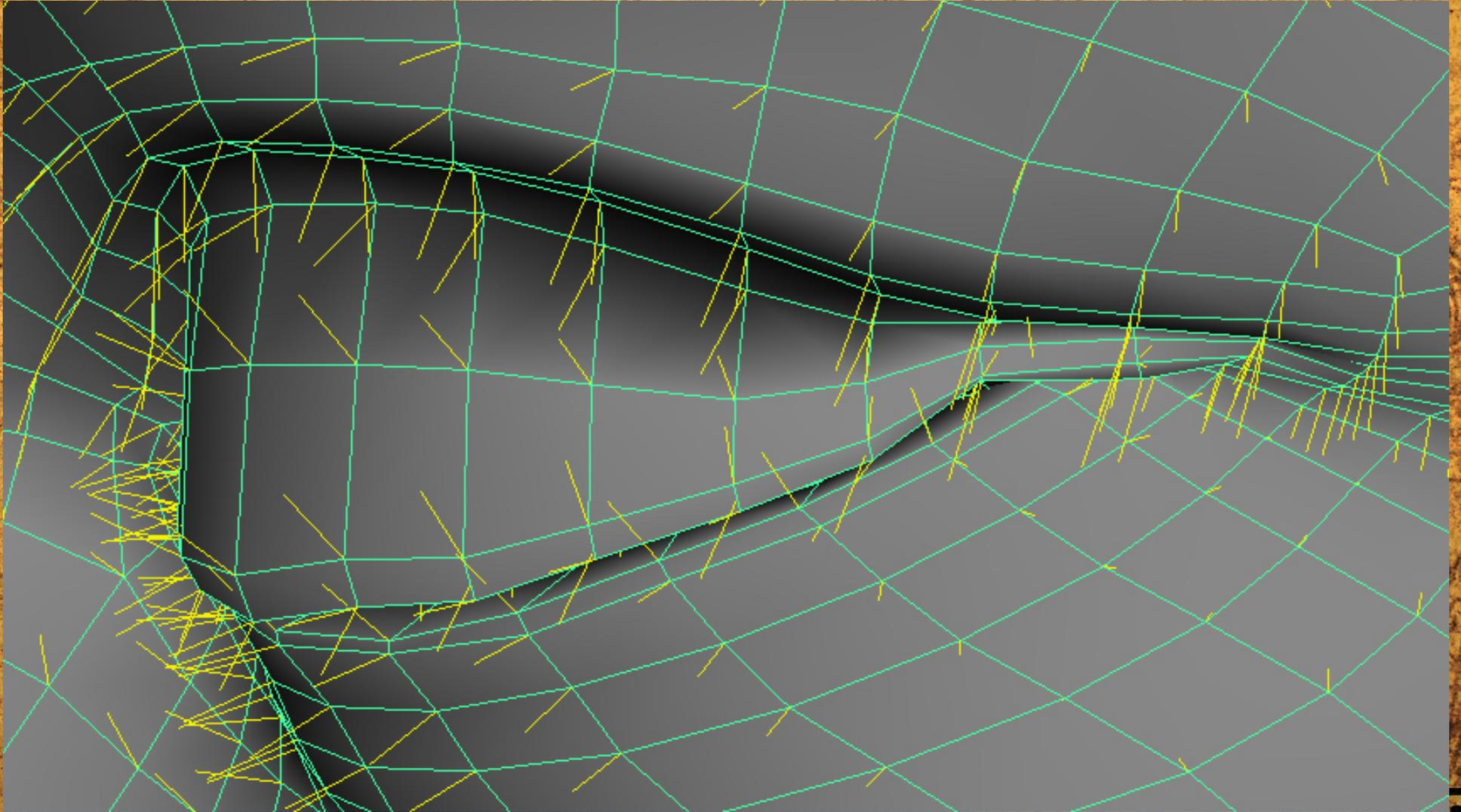
*-Only joint rotation modifies the vertex normal*

*Tried some tricks which ultimately did not work.*

# Normals not transforming based on translation for the eyes:

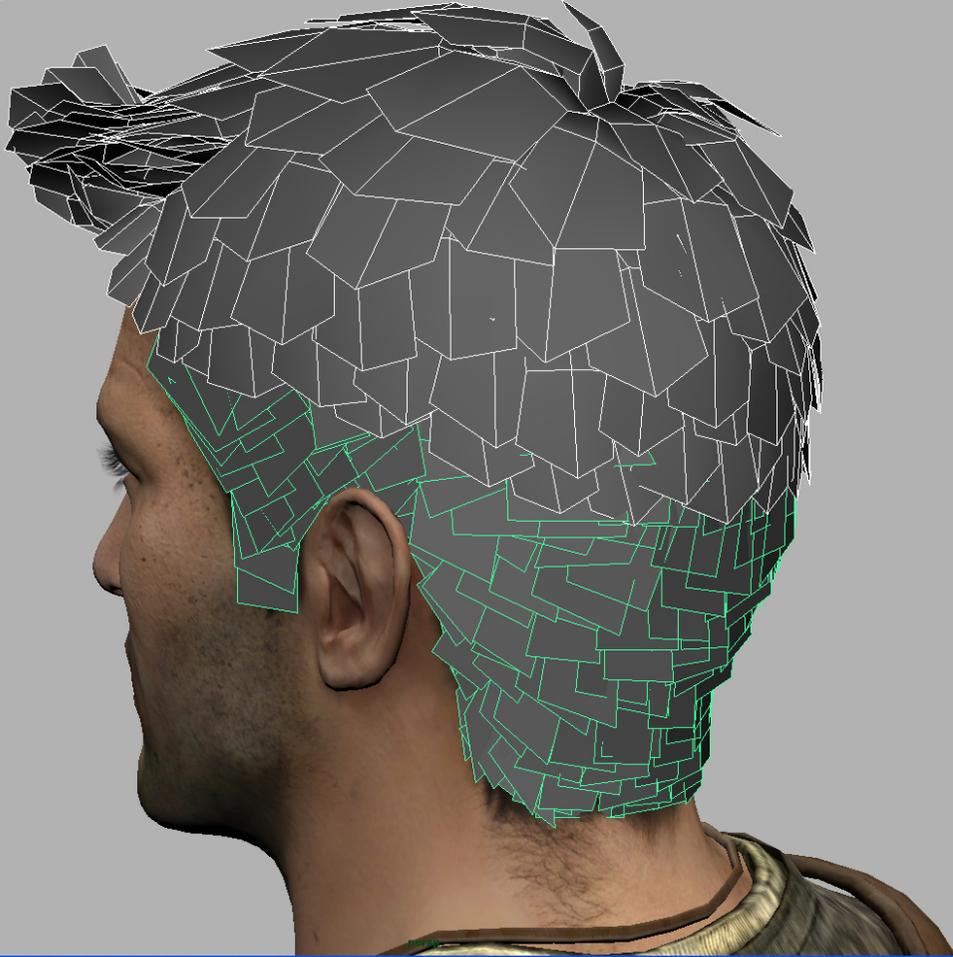


# Normals not transforming based on translation for the eyes:



# Improved Hair:

U1 2308 tris



U2 4002 tris



# Improved Hair:

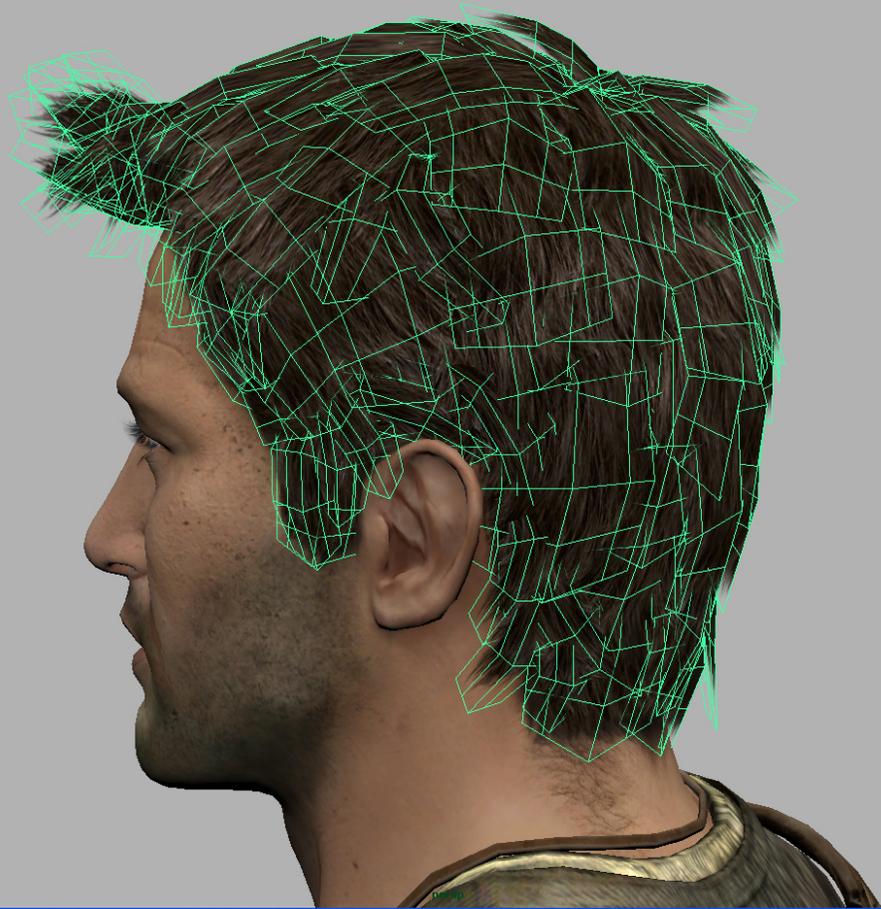
U1



U2



# Improved Hair:



# Outsourcing:

# Outsourcing Characters:

- Turned out to be more work for high quality assets but was useful for smaller assets
- When things did work it allowed us to focus on the more important stuff
- We need to take the time up-front to really evaluate the companies properly
- Communication was tough at times

# Conclusion:

- **Better way to deal with LODs**
- **Possibly get rid of arbitrary mesh**
- **Manage actors in an easier way**
  - multiple costumes, texture sets, etc..
- **Reference skinning**
  - figure out a way to reference skinning information so you only update in one place
- **Eyes**
  - Eye's still feel creepy at time and don't always match the environment
- **Tackling optimization earlier**
- **Find better solution for multiplayer faces and skeleton variation**

Thanks!



# Q & A:

Email:

Rich "Custom" Diamant  
[rich@rd3d.com](mailto:rich@rd3d.com)

Judd "Oiiii" Simantov  
[juddsim@gmail.com](mailto:juddsim@gmail.com)