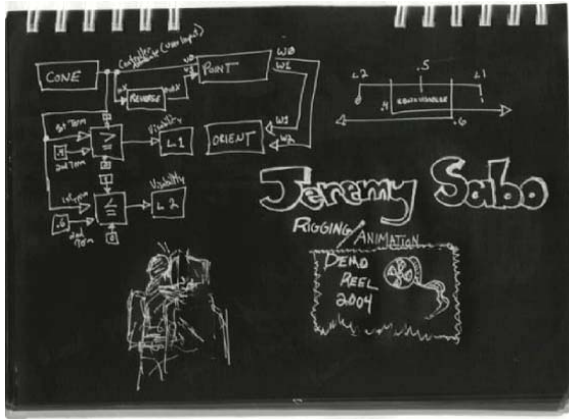


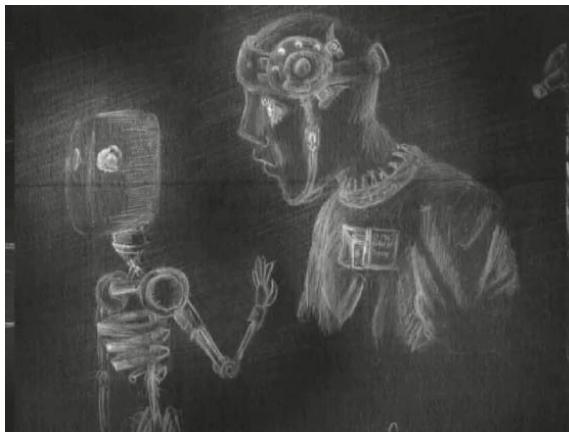
Jeremy Sabo (systemshutdown.com)

Demo Reel Outline

Aug 2004

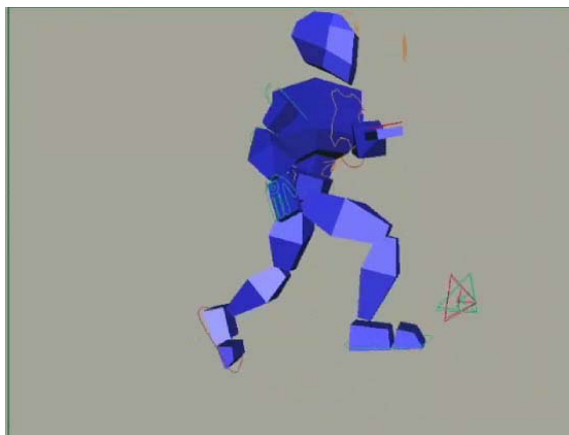


Title Screen
(0:00 - 0:04)



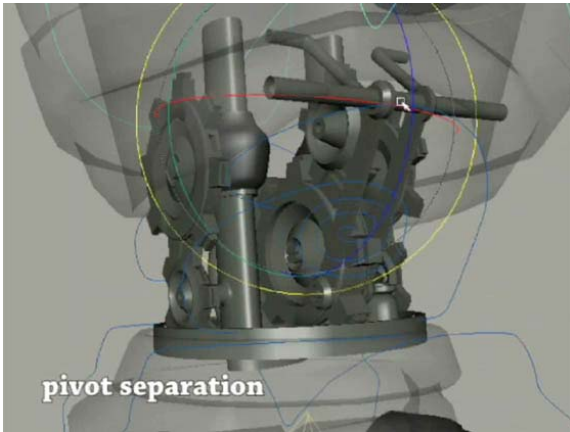
Luma Concept Sketches
(0:04 - 0:12)

Luma was an idea that started before I had ever even touched Maya. The simple sketch slowly changed into a 3D model and, eventually, into a short story I plan on making one day. These are scans made from my sketchbooks, over the last two years.



Luma Proxy Vs High Res Geo
(0:04 - 0:12)

Although the low-poly version of the Luma model runs fairly quickly, I created this block proxy version to ensure the quickest feedback possible when animating.



Luma Gears Rig

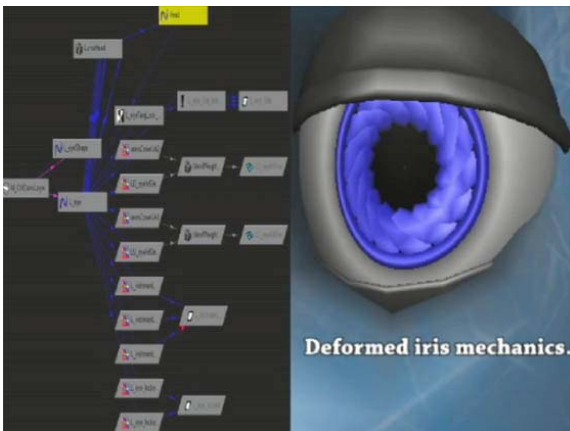
(0:17 – 0:47)

I wanted to go against popular suggestion and put a robot in my demo reel. However, I wanted it to stand out among the rest. I planned on all limbs to be connected by gears that are automatically driven when an animator manipulates the rig. The toughest challenge was separating the rotation pivot points. For

example, when the animator selects the chest control curve and twists the chest, the hip twist is a pivot point that is below and further back than the point in which the chest rotates backwards and forwards. This became even more complex in order to allow the animator to alternatively select the hips and manipulate the rig from the bottom up.

The shoulder provided me with a different headache, although it originally seemed simple. The rotations needed to be separated, yet controlled via FK or IK. Setting this up via FK doesn't provide much of a challenge, but separating the shoulder's up/down rotation from its twist and backwards/forwards rotation, yet having it still work with IK was not so simple. In the end, I had to create separate "base" rigs that would contain the IK solver, and then I created a separate rig that remained constrained to the mechanical limits of the gears, yet does its best to also constrain itself to the base IK solver arm.

The neck was (luckily) simple. It just required a little bit of creative grouping and hierarchy organization to separate the rotations, something I was quite comfortable with after working on the torso gears. You will notice that the head and hand controls are located in odd locations, there's a reason for this, which I will go into later.



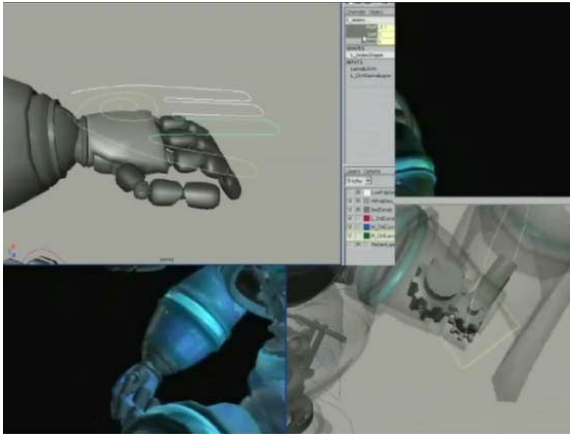
Luma Eyes

(0:47 – 0:58)

I wanted the robot to have an iris that dilated like a camera shutter. There were some fairly simple solutions available, but many of them didn't seem to work if you then deformed the rig, as I needed to do to get it to fit into the egg-shaped eye socket. I finally came up with this solution of using point/aim constraints w/ locators.

I then just put the locators and geometry into the same group, slapped a lattice over the top and vwalah!

The eyes can be controlled via a target curve or the eye curves found under the head rig. I created this separate method to allow control over the eyes, yet have the eyes remained still in their socket in case Luma is ever powered down.



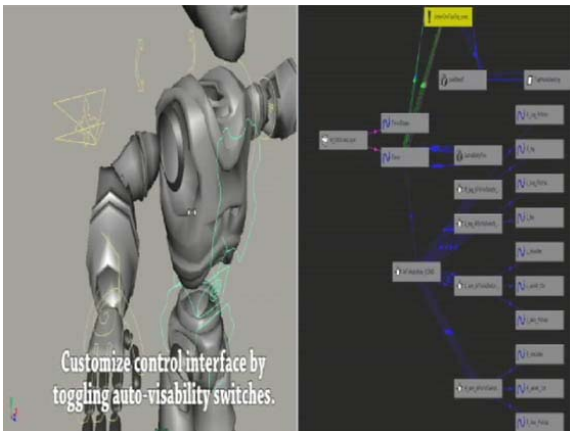
Misc. Details

(0:58 – 1:06)

Hand controls are fairly standard. However, I personally use a “spread”, “point” and “curl” method. Most rigs seem to just have the fingers curl, which doesn’t allow the fingers to point straight from the knuckle.

There are also the gears in the elbows and knees that were created using direct connects, w/

reverse nodes, from the joint rotations. I’m a big fan of little details in order to make something feel more realistic. I chose gears, because I’d already seen too many hydraulic robots, and I feel that the point/aim constrained solution had already been done, and why beat a dead horse?



Customizable Interfaces

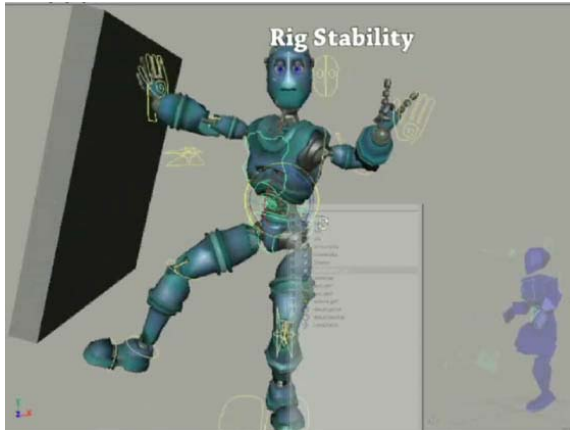
(1:06 – 1:11)

Depending how the animator works, I’ve allowed customization of the control curves. For example, the animator can either have the IK curves disappear when the FK rig has been switched on, or visa versa. Or, the auto toggle can be switched off, which leaves both IK & FK rigs visible. It was very important to me to allow this

customization, yet keeping the rig responding in real time, so all these options were created with Maya’s Hypershade nodes. I created my own IK/FK switching rigs because it gave me more control over it’s behavior.

As I said earlier, some of the controls are in odd locations. This is because the animator can put these controls wherever he/she prefers to have them. The animator can translate these controls to any convenient location for animating details (fingers and facial animation), which seems to help significantly, especially when zoomed in. The controls are still attached to the rig, so they won’t be left behind when the rig is animated.

I try to keep most of my controls 3D view for efficiency reasons. I believe the controls should be right where the manipulator will appear when the control is selected, which also decreases the possibility of Carpal Tunnel.



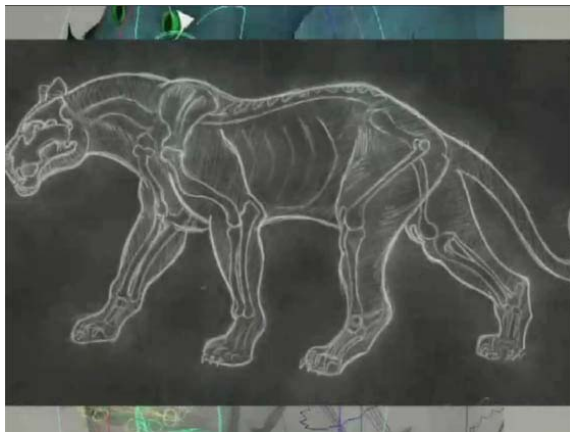
Rig Stability

(1:11 – 1:22)

One arm and one leg is switched to IK, the other arm and leg are set to FK. I just wanted to show that the rig doesn't fall apart when manipulated. Also, even on an AMD Athlon 2000 CPU, with a screen capture program running, it still runs in real time.

The window in the bottom right is an animation cycle playing

on the rig, while I'm translating and rotating the top level control curve group that contains all the rigs animation controls.

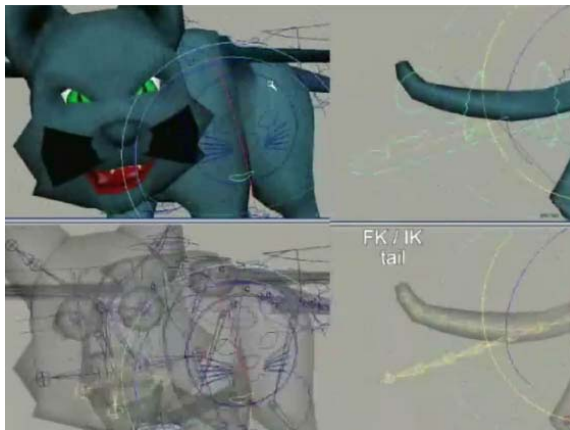


Feline Anatomy Study

(1:22 – 1:26)

I was told that I should show some diversity in my rigs, like showing a quadruped or a bird, so I made a flying kitty. After watching a few videos, searching the internet, and drawing a few (included) study sketches I had a fairly deep understanding of the bone structure and animation of a cat. The shoulders were the most

tricky to get right. Although, I should point out that I do realize that by attaching the wings on top of the existing shoulders, it's no longer a quadruped.



Flying Kitty Rig

(1:26 – 1:50)

The facial animation deformation is controlled entirely with bones. The iris dilation effect was a blend shape. The ears and whiskers can be flopped around by translating the control curves, while the mouth uses rotation (which feels more intuitive). As the

jaw opens, the corners of the mouth spread open using SDKs.

The tail can be controlled with FK (2 joints per control curve) or spline IK. The base of the tail can be locked or free, allowing the tail to appear loose or stiff when the rear is moved around.

The wing controls I kept simple, intuitive and fun. Translating the wings up and down bows the wings as if air is pressing into the skin as they flap. Translating the wings backwards and forwards extends and contracts the wings. Rotating simply rotates the entire wing rig. I noticed that with this setup, I was able to create flying animation cycles in seconds.



Kyle Facial Rig

(1:50 – 2:23)

The Kyle geometry was given to me by a friend who needed a rig with more control than the one that had been originally built for the character. The rig originally had all FK with visible selection handles for the limbs and torso, and blend shape targets for the face. I returned the character to my friend with a rig that contained

a stretchy spine, IK/FK arms and legs with free/locking wrists, shoulders, eyes and head, and many other bells and whistles. A little while later, I was happy to hear that he found my rig “fun” to work with. He lives in Belgium, so we stayed in contact via email as he provided me with constant feedback. I also received permission to use the character in my demo reel.

Eyelid shapes created via blend shapes, all other deformation created with bones. However, I’ve also put in “kiss” and “puff” attributes that were created with blend shapes. The deformation between the eyebrows was actually created using bones attached to curves via motion paths, which had to then be modified in order for me to control them through an attribute, instead of time. It gave me that nice little swoop of the eyebrows that I was looking for.

The tongue control was using an IK spline curve and tweaked with a blend shape to retain volume.



10 Second Club Entry

(2:24 – 2:36)

I created this animation for the 10 Second Club in 3 days, less than 15 hours total and was very satisfied with the results of this

exercise. The rig was responsible for the speed in which I completed this animation. Also, notice how the control curves themselves deform when translated. At first I thought I might have been over-detailing this part, but while working with the rig, I found it very intuitive to work with.



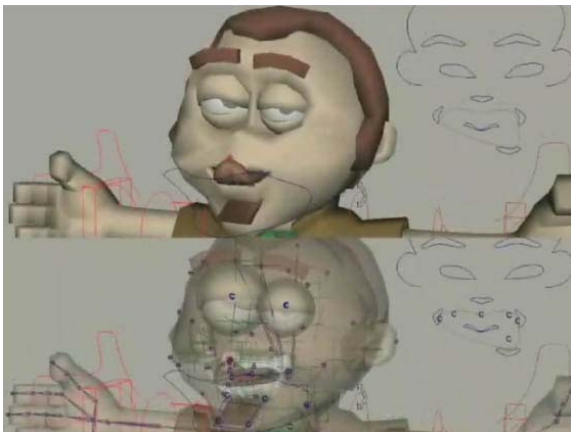
Credits and Misc. Clips (2:36 – 2:47)

The upper left video shows the infamous “Ball / Door Challenge”. I had created a way, using entirely Hypershade nodes, to have a ball pass through a door, pushing it open as it passed. However, if the door closed behind the ball and the ball then changed direction, moving back through the door, it would push the door open the other direction. The video makes it easier to explain). This was an exercise in how to “code” using condition nodes, use attributes as variables, as well as how to manipulate the data feeding in and out of a Set Driven Key node.

The upper right is a time lapse of the kitty being modeled.

Lower left is me playing with the proxy rig.

Lower right is a time lapse demo of me animating the facial animation for my 10 Second Club entry. All the lip-syncing was done in less than 4.5 hours. I plan on making a facial pose tool to speed this process up even more. My problem with current pose tools, is that they force the face entirely back to a preset pose shape. So I ask, “What if the character is constantly squinting with the corner of his mouth stuck in a snarl?” It would be annoying to have to reset this over and over again after using the preset poses. The tool I have planned will be a great exercise in building a complex UI. Stay tuned!



Bonus Animation (2:47 – 2:52)

While crunching to have this demo reel completed for Siggraph 2004, I took a 45 minute break to make this silly little animation to demo the mouth and tongue.

All elements were created in Maya 5, composited in After Effects and edited in Premiere.