

**FTV118918**

## **Rigging Mechanical Objects in 3ds Max**

George Maestri

LinkedIn Learning / Lynda.com

### **Learning Objectives**

- Learn how to use scripts and expressions to automate rigging
- Learn how to use character animation tools to rig mechanical objects
- Learn how to automate the animation of a complex assembly
- Learn how to create a control panel to manipulate animation

### **Description**

Learn techniques for rigging mechanical objects, vehicles, machines, and other inanimate objects in 3ds Max software. Many people normally use 3ds Max software's rigging tools for characters, but you can also use them to rig things such as vehicles, motors, pistons, assemblies, conveyor belts, and more. This course will demonstrate practical applications of 3ds Max software's extensive rigging tools to automate the animation of a variety of mechanical objects and systems.

### **Speaker(s)**

**George Maestri** is an animation industry professional with extensive experience as a writer, director, and producer, working on a number of hit television shows, including South Park and Rocko's Modern Life. As an entrepreneur, he built a successful studio that produced several hundred cartoons for broadcast, education, and film. George has written over a dozen books on computer graphics. As an educator, he has taught at several top animation schools, including California Institute of the Arts. He currently authors and develops CAD and 3D content for LinkedIn Learning and Lynda.com

## **Mechanical Rigging in 3ds Max**

### **What is Rigging?**

Rigging is a process that makes complex assemblies of objects easier to animate and manage. Rigging is most commonly associated with character animation, but the same tools directly apply to complex mechanical objects and assemblies.

### **A good rig will do the following :**

#### **Automate as much as possible**

If parts of the assembly always move in relation to other parts, then these can be automated to cut down on animator overhead. Gears in a clock can be rigged to turn in relation to each other, pistons can automatically compress/expand, and so on...

#### **Create Realistic Movement**

Parts in an assembly should move realistically and stay in proper alignment

#### **Be Animator Friendly**

A good rig will abstract the animation to a collection of controls that the animator will manipulate. The goal is to have the animator work the controls and never have to touch the mesh.

#### **Easy to Understand**

Rigging is essentially a computer interface, so good interface guidelines should apply. Make controls clear and easy to understand, provide labels if needed, and color code key parts of the rig.

## Designing and Creating a Rig

Designing and creating a rig is similar to creating a user interface. When considering how to rig an assembly, you need to go through a general iterative design process:

### **Understand How the Object Moves**

When creating a rig for a mechanical object, you need to know how it should move. You need to know exactly how the parts are connected and how they interact. This may require research about the mechanics of the object, and you may have to talk to engineers or designers. You also need to know limitations of the object's motion – how far is it supposed to bend? Does the dial go in increments or is it continuous? Finally, you should also consider how the object will be presented. If a particular action is never shown in the presentation, you may not need to rig it.

### **Decide what needs to be controlled**

As the rigger, you need to make decisions about what can and cannot be animated in an object or assembly. A hydraulic arm may simply need to be positioned in space, so you can give the animator a single position control and rig the rest. Something more complex, such as an airplane, may need additional controls for things like landing gear and propeller rotation.

### **Determine what can be automated**

The rig is supposed to make life easier for the animator. This means the rig should control much of the behavior of the assembly. Hydraulic pistons can automatically expand/contract, hoses can bend automatically, gear assemblies can rotate in sync, and so on. Automating these things is where rigging really matters to an animator. Be sure to understand the animator's role and what needs to be controlled before rigging.

### **Make it animator friendly**

Your rig is the user interface for the scene and the objects being animated. Make that interface easy to use. Observe good interface design practices with your rigs. Make controls easy to spot, use shapes to define function, use color as a guide, and be sure to add labels, if needed.

### **Keep it Simple**

As with any good interface design, simpler is almost always better. Keep the controls direct and easy to understand. Don't add controls for things that do not need to be animated. Make the resulting animation simple to control as well. You don't want a spaghetti bowl of animation curves for the animator to wade through.

### **Test until it doesn't break**

Think of every possible way to move the rig and run it through its paces before handing it over to the animator. Make sure the rig is rock solid.

## Creating Controls

When building a rig, we need to create controls that are animator friendly. These controls are typically created using non-renderable objects such as curves and text objects. Proxy objects can also be used as controls. These controls should be easy to understand and placed where they are easy to access. Naming schemes and color schemes can also help with readability.

### **On-object controls**

These are controls that are placed on/near the objects being controlled. The scoop of a digging machine may have a rotational control directly above the scoop.

### **Multi-Purpose controls**

Some controls may be easier to use if they control more than one parameter. In the above example, the rotational control for the scoop could be used to determine the position of the digger's arm.

### **Control Panels**

You may need to abstract and/or consolidate controls into a single control panel. This is often useful for assemblies where there are a lot of parameters to manage.

In general, the animator should always be operating the controls and not the object itself. This enables the mesh of the object to be frozen so that is not tampered with, moved out of alignment, deleted, and so on...

## Rigging Tools

3ds Max has a number of rigging tools. Some are very easy to use, others are more complex. Typically, we start with the simpler tools and work towards the more complex ones in the rig. Generally, the goal is to keep the rig as simple as possible, so simpler tools tend to fit this goal. Don't make things more complicated than necessary just because you have a lot of really cool tools in your arsenal. Other people may have to sue/debug your rigs, so easier to understand rigs make sense.

That said, more complex tools can provide a lot of functionality and they should absolutely be used when needed.

## Basic Four Rigging Tools

These rigging tools are the simplest, but many times, they are all you need. If you learn these, you'll be able to do a lot of basic rigging.

### Pivots

These simply define the axis of rotation and scale for any object. Placement of this axis is critical for anything involving rotation. Centering the pivot may not align directly to the volumetric center of the object. If you need multiple pivots, then you must create helper objects such as a dummy or point to position the other axis of rotation.

### Links/Hierarchies

This simply connects objects together in a hierarchy. The structure of the hierarchy usually reflects how the object will be animated, but novel hierarchies can also be used. Many times, we may link helper objects or animation controls into a hierarchy to make it easier to animate.

### Inverse Kinematics

Allows objects to rotate automatically to meet a goal object. Traditionally used in character animation, it can be used for all sorts of mechanical assemblies, from robotic armatures to pistons. Several types of IK are available

**HI (History-Independent)** – Normally used for characters and longer sequences, it allows for the IK solution to be calculated on every frame. It does basic joint rotation but no sliding joints

**HI (History-Dependent)** - The HD Solver is a solver well-suited to use for animating machines, especially ones with sliding parts that require IK animation. It lets you set up joint limits and precedence. It has performance problems on long sequences, so ideally use it on short animation sequences. It is good for animating machines, especially ones with sliding parts.

**IK Limb** – Only works for two-bone chains. Fast to use, but limited in application.

**Spline IK** – Allows a spline curve to control the orientation of the joints/objects. It is good for hoses, springs, and other flexible objects.

### Constraints

A method of connecting parameters together outside of traditional hierarchies. Constraints can be used to align an object's position, rotation, or scale to another. They can also be used to point objects at one another, and attach them to paths or surfaces.

## **Extended Rigging Tools/Concepts**

3ds Max has a number of sophisticated rigging tools that go beyond the standard four. These can introduce a finer degree of control over your rigs and add much more sophisticated behaviors.

### **Animation Controllers**

These are modules that control an object's motion, and are really the foundation of all animation in 3ds Max. Controllers can use standard keyframing techniques to create motion, but they can also be used to connect objects together using algorithms, procedures, or other processes.

**Morphs** – Morphs are used to control shape. They can be used for flexible objects such as springs and hoses.

### **Wire Parameters**

Wire Parameters lets you link any two object parameters in the viewport, so that adjusting one parameter changes the other automatically. This enables you to set up one- and two-way connections between specified object parameters, or to control any number of objects with a dummy object containing the desired parameters. By wiring parameters, you can set up custom constraints directly without having to go to Track View and assign controllers.

You can also add expressions to Wired Parameters to introduce mathematics and logic to object connections.

### **Expressions**

If you want to dig even deeper, you can use the Expression Controller to more discretely control objects using math, logic, and basic coding.

### **Reaction Manager**

A controller that links the behavior of two objects with user-defined curves that can create sophisticated behavior not available with standard expressions. The controller works by linking a Master parameter to a Slave parameter. Moving or adjusting the Master affects the slave in a non-linear way. The link is controlled by an animation curve that specifies behavior at all points throughout the range of motions.