Tutorials: Animation

Autodesk® 3ds Max®

2010

Autodesk®

Autodesk® 3ds® Max 2010 Software

Trademarks

Disclaimer

Animation Tutorials

4

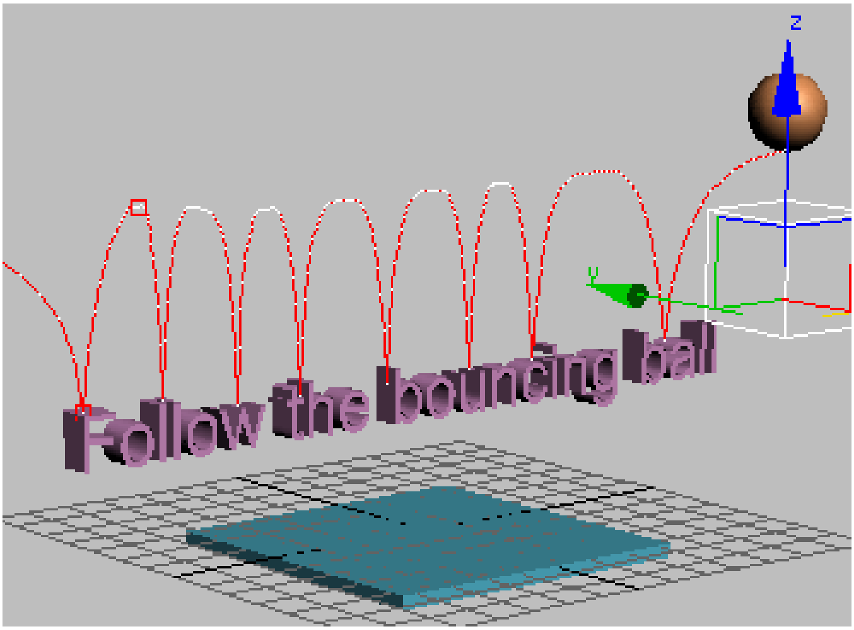
Here you can learn the fundamentals of creating animation with 3ds Max.

Features Covered in This Section

- Auto Key animation
- Ghosting
- Dummy objects and animation
- Track View – Dope Sheet
- Looping animation
- Weighted list controllers
- Set Key Animation
- Path constraint and Path deform animation methods
- Schematic View Improvements
- Animation control through List Controllers
- Scripting
- Wiring and expressions
- Sound track editing

Animating with Auto Key: Bouncing a Ball

A bouncing ball is a common first project for new animators. This classic example is an excellent tool for explaining basic animation processes in 3ds Max.



In this tutorial, you will learn how to:

- Create animation using transforms.
- Cop keys in the track bar.
- Use ghosting to visualize in-betweens.
- Use tangent handles in the Function Curve Editor to control in-betweens.
- Create looping animation using Parameter Out-of-Range Types.
- Animate using dummy objects.
- Use Layout mode.
- Apply Multiplier curves.
- Work with the Dope Sheet Editor to speed up animation and reverse time.

- Animate using Set Key mode.

Skill level: Beginner

Time to complete: 1 hour 45 minutes

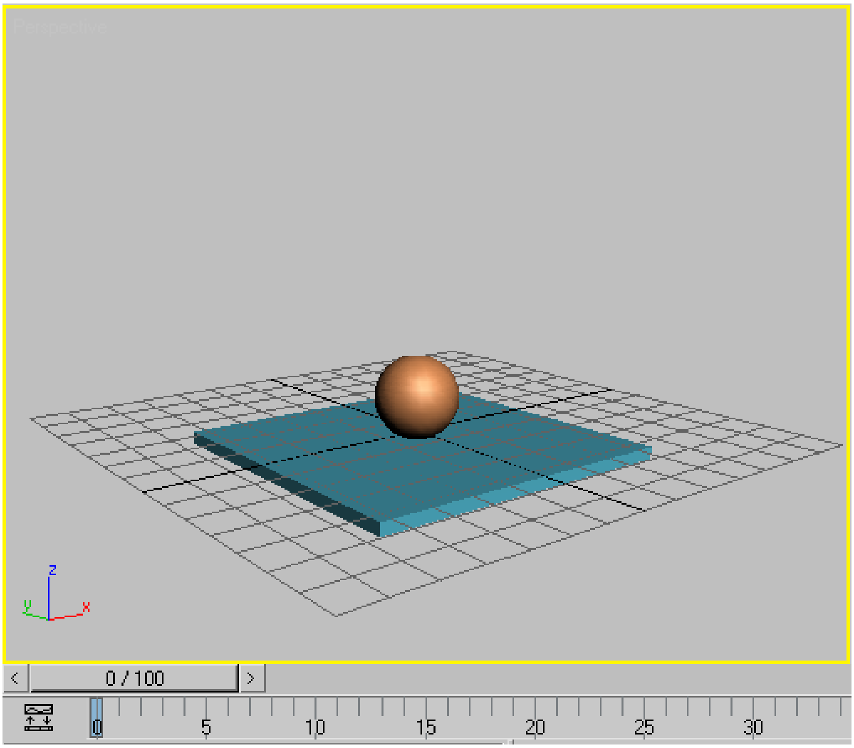
Creating Animation Using Auto Key

In this lesson, you'll start learning how to animate in 3ds Max.

Animate the ball using the Move transform:

1  On the Quick Access toolbar, click the Open File button, navigate to the *animation\auto_key* folder and open *bounce_start.max*.

NOTE The following illustrations display the grid differently than you will see in your viewports. For ease of use, press G on the keyboard to hide the display of your grid.



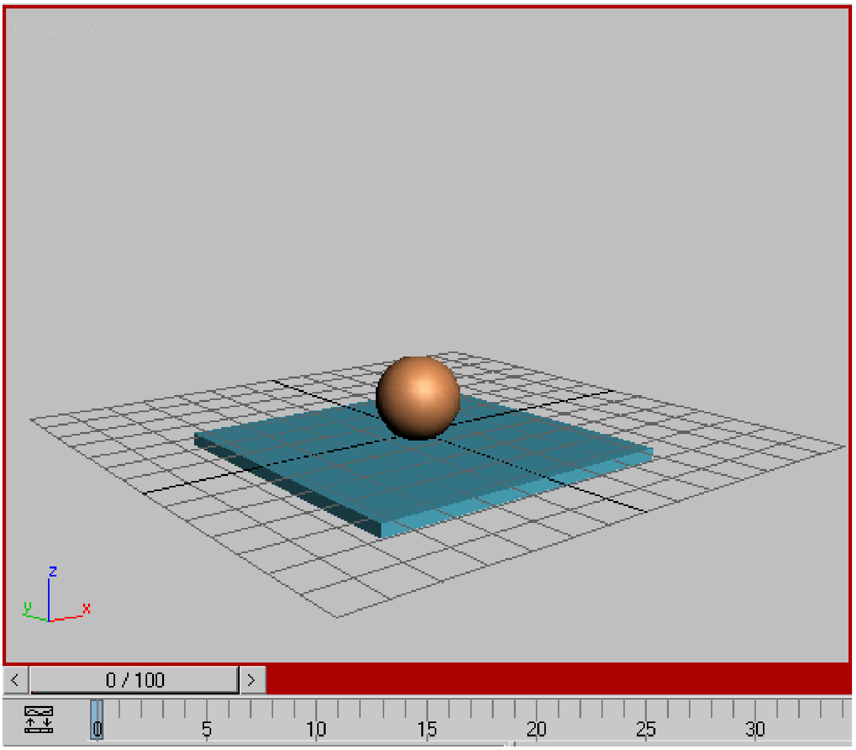
Perspective viewport: bounce_start.max

This file has the Perspective viewport displayed on the lower left. This is not the standard viewport layout; the layout has been customized for this lesson.

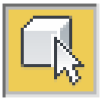
2 

Click Auto Key to turn on this feature.

The Auto Key button and the time slider background turn red to indicate that you are in animation mode. The viewport is outlined in red, as well. Now, when you move, rotate, or scale an object, you create keyframes automatically.

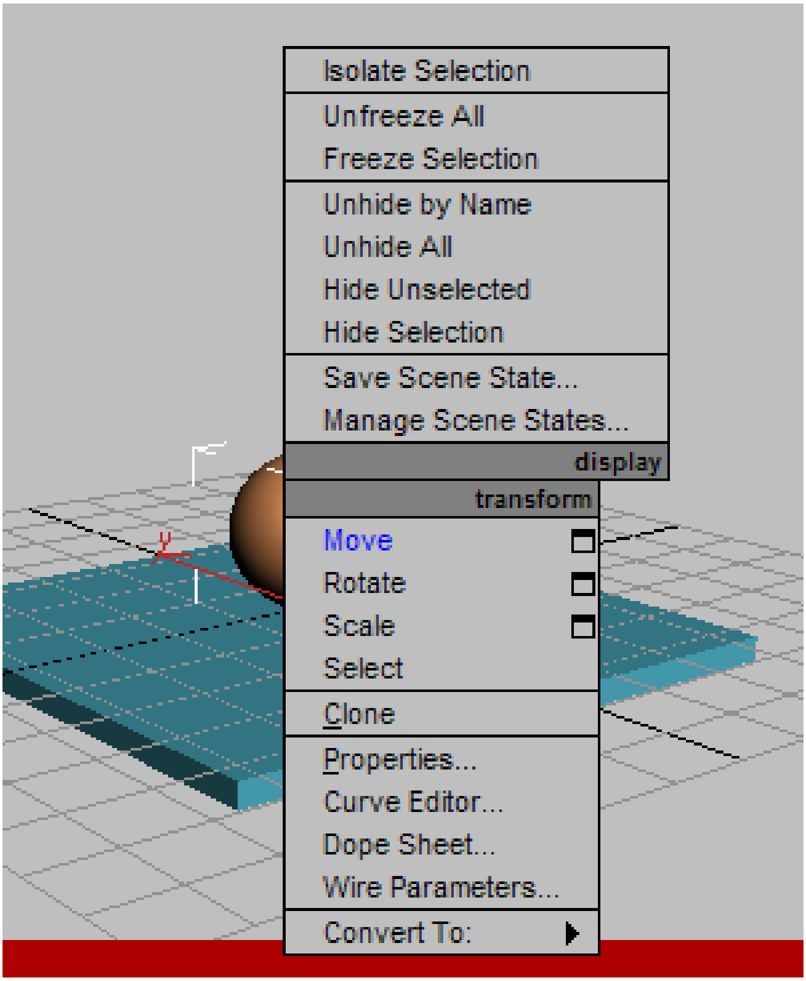


Time Slider background turns red

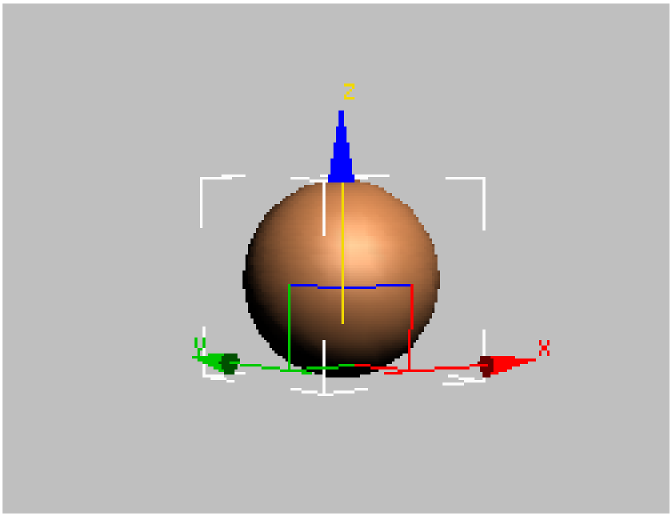
3  Click to select the ball in the Perspective viewport.

It is displayed within white selection brackets, indicating that it's selected.

4 Right-click the ball and choose Move from the Transform quadrant of the quad menu.



The Transform gizmo appears in the viewport. The Transform gizmo lets you easily perform constrained movements. As you move your cursor over the Transform gizmo, the different axes and their labels turn yellow.



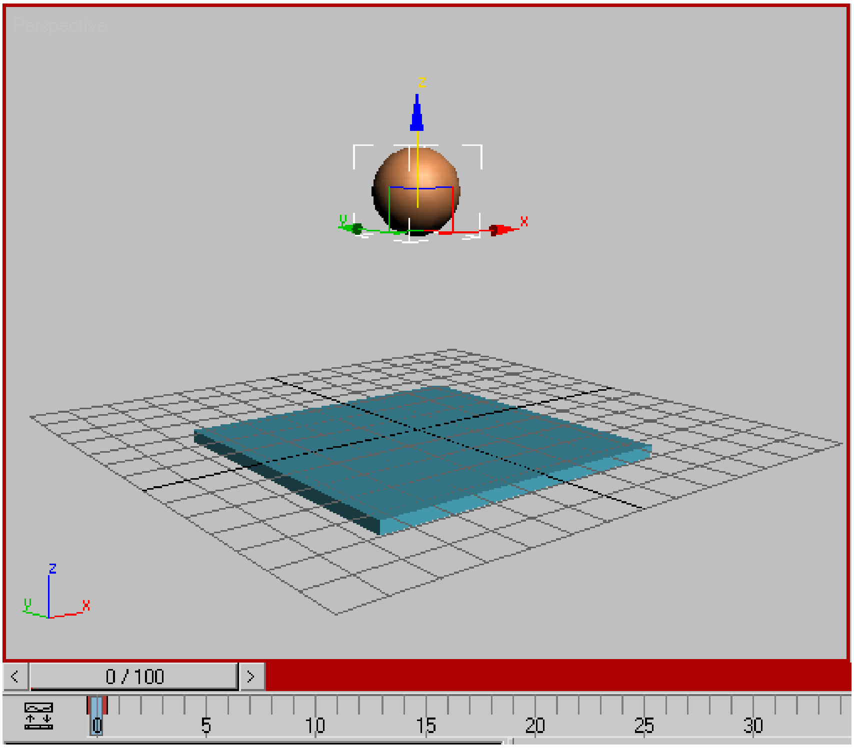
5 Position the mouse cursor over the Z axis, and when it turns yellow, click and drag upward to raise up the ball in the air.

As you move the ball up in the air, notice the Z value changing in the Coordinate Display below the track bar.

The ball's position at frame 0 is now fixed above the box.

A Position key is created when you do this. The key is displayed on the track bar.

The track bar displays the keys for whatever is selected in the viewport. The track bar is found directly beneath the time slider and above the prompt area.



Keyframe created in Auto Key Mode

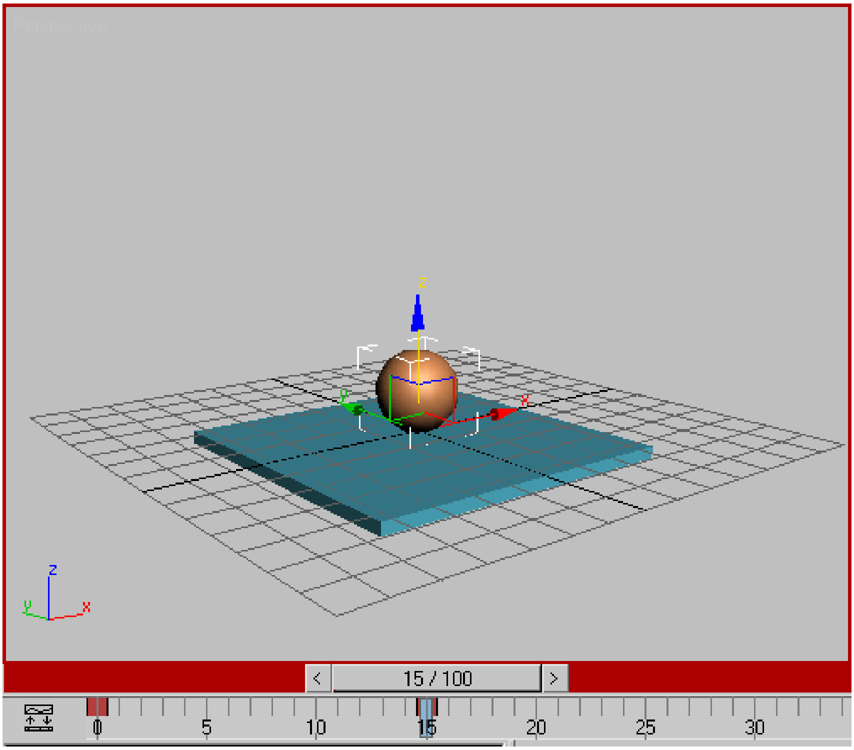
6 Move the time slider to frame 15.

To move the ball precisely down to the table surface, put your cursor in the Coordinate display Z field, and change the value to 0.

Coordinate Display type-in for precision animation.

The box is built off the world origin, so a Z position value of 0 will set the ball directly in contact with the box.

NOTE The World Origin, (0,0,0) in XYZ coordinates, is shown by the intersection of the dark lines on the Home grid.



Frame 15: Ball is in contact with the box

You need to make the ball rise up to its original position at frame 30. Instead of moving to frame 30 and moving the ball back up in the air, you'll use a different method.

TIP You can zero the Z coordinate (or any other spinner) by right-clicking the spinner arrows.

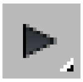
---

7. Put your mouse over the time slider's frame indicator (the grey box that currently reads 15/100 and right-click.

   The Create Key dialog appears.

8. In the Create Key dialog, change the Source Time to **1** and the Destination Time to **30**, then click OK.

   This copies the key from frame 1 to frame 30.

9  Click Play Animation to play the animation, or drag the time slider back and forth between frames 1and 30.

The ball moves down and up between frames 1 and 30, and stays up in the air between frames 30 and 100.

10  If you clicked Play Animation, click Stop (the same button) to end the playback.

Next, you will set the length of the active time segment to 30 frames.

11  In the time controls, click Time Configuration.

12 In the Time Configuration dialog > Animation group, set Start Time to **1** and End Time to **30**. *Don't* click the Re-Scale Time button. Click OK.

3ds Max lets you work in an active time segment that's a part of a larger animation. Here you are making frames 0 through 30 the active time segment. Notice that the time slider now shows only these frames. The other frames still exist, they just aren't part of the active segment at the moment.

13 Play the animation.

The ball goes up and down. Since the first and last frames are the same, the animation appears to cycle as it plays.

The ball moves, but it doesn't have "bounce" yet.

14 Stop the animation playback.

3ds Max made decisions on how the in-betweens are being distributed. Right now they are evenly distributed so the ball has no acceleration. It doesn't speed up or slow down; it just floats along with no sense of weight.

You need to simulate the effect of gravity so that the ball slows to a stop at the top of its bounce, speeds up as it approaches the table, and then bounces up again. To accomplish this, you'll use the key interpolation curves available on the Curve Editor. You'll also use the Ghosting feature to help visualize what the interpolation curves are doing.

Controlling In-Betweens

To make the ball bounce more convincingly, you'll change the interpolation on the key at frame 15.

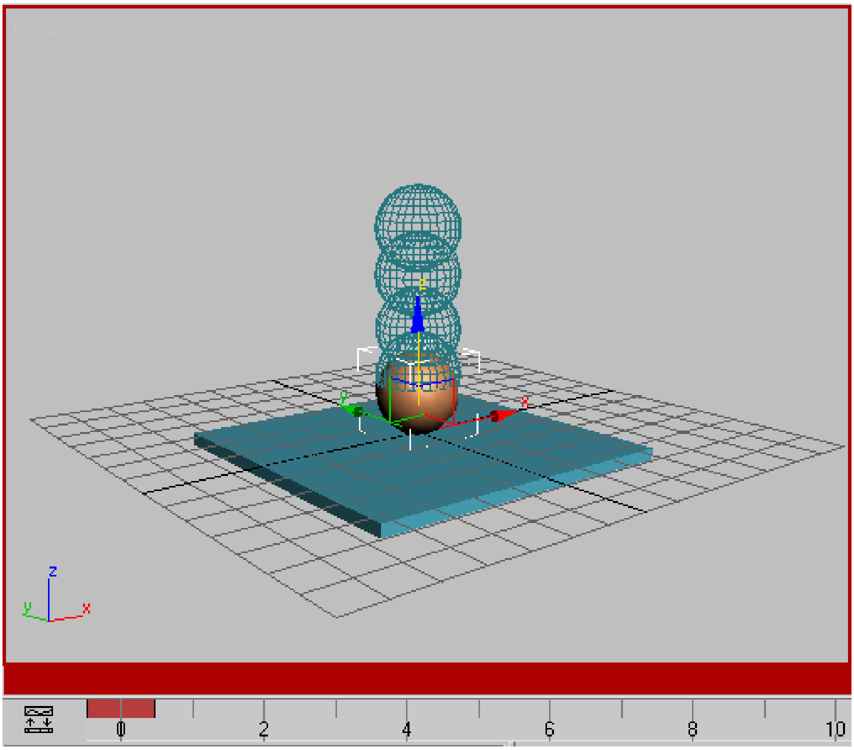
You'll use the tangency handles available in the Curve Editor. The tangency of the curve will determine the position in space of the in-between frames. Ghosting will let you see where the in-betweens are being positioned.

Using ghosting to visualize in-betweens:

1. Move the time slider to frame 15.
2. From the Views menu, click Show Ghosting to turn on this feature.

   The ghosting feature shows object positions before the current keyframe in an aqua color.
3. Go to Customize menu > Preferences > Viewports tab, and set Ghosting Frames to **4** and set Display Nth frame to **3**. Click OK to exit the dialog.

   The viewport displays the ghosting.



Ghosting shows the object positions on previous keyframes.

4 Play the animation, and then stop.

5 Now, to control the in-betweens, right-click the ball in the viewport and choose Curve Editor.

The Function Curve Editor is displayed across the top two viewports.

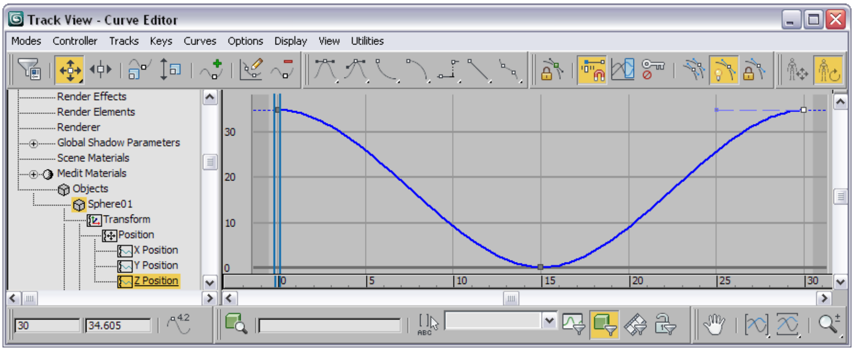
The Curve Editor is composed of two windows, a Controller window on the left that shows the names of tracks, and a Keys window on the right that shows keys and curves.

6 In the Controller window on the left, click to select only the Z position track.

NOTE If you don't see the Z Position track, click the plus icon to the left of the ball to expand the ball’s tracks. If you don’t see a plus icon, right-click and choose Manual Navigation, hold down Alt and right-click, and then choose Expand from the quad menu.

There are many possible configurations in the Curve Editor, so you may encounter variations from the standard.

Now the only curve displayed in the Keys window is the one you want to work on.



Function curve for bouncing ball Z position

7 Move the Track View time slider (the double aqua line in the Key window).

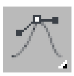
As you move back and forth the animation plays in the viewport.

If you look carefully you'll see a dark dot on the curve at frame 15.

8 Drag around the dark dot (the position key) to select it.

The selected key turns white on the curve.

You will now manipulate the curve using tangency handles. To access the handles, you must change the tangency type.

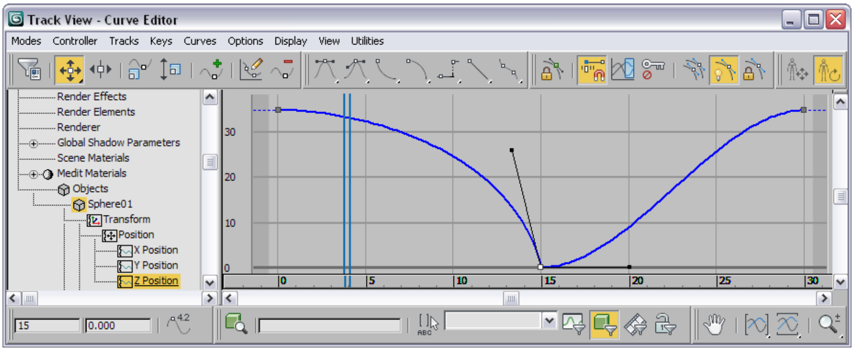
9  On the Track View toolbar, click Set Tangents To Custom.

If you look carefully, you'll see a pair of black tangency handles have appeared on the curve.

10 Hold down the Shift key and drag the left handle on the left upwards in the Key window.

Using the Shift key lets you manipulate the left handle independently from the right.

The curve now looks like this:



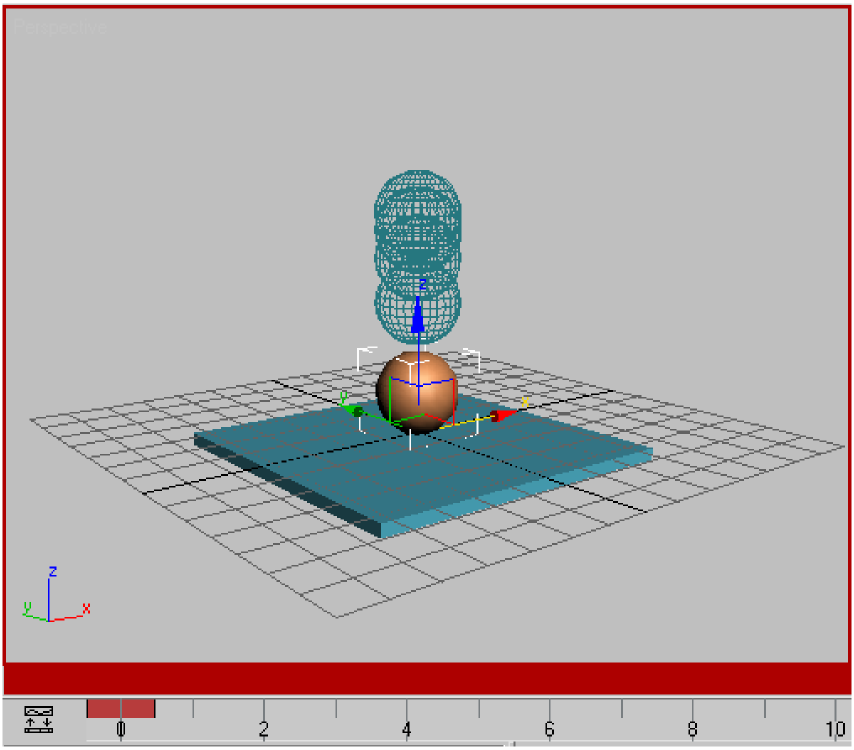
Shift key allows for manipulation of individual handles.

You'll change one more setting to make this tool more useful.

Using Interactive Update:

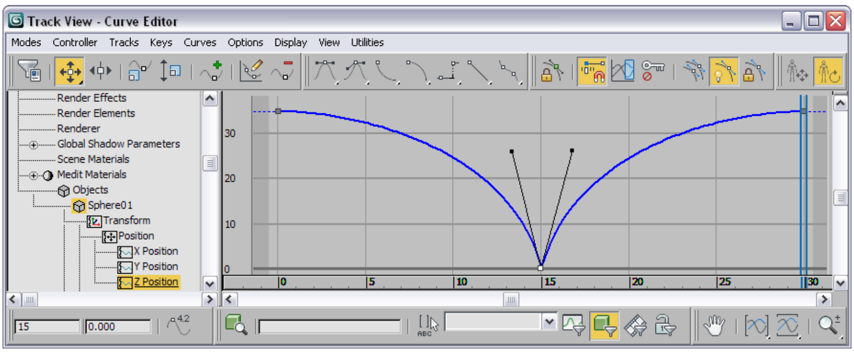
1 On the Track View Options menu turn on Interactive Update. Now move the time slider to frame 15, then manipulate the tangency handle while observing the effect in the ghosting. You can clearly see the changes as you work.

2 Set the tangency handle so the in-betweens are mostly drawn toward the raised position (see the illustration that follows). With interactive update on, you can do this with very fine control.



Interactive update and ghosting

3 Move the time slider to frame 30, then adjust the right tangency handle so it approximately matches the left one.



By manipulating this handle you can get different effects. The upward movement of the ball as it bounces off the table will determine the perception of the weight of the ball. The ball will appear to be bouncy, like a tennis ball, if the two handles are similar. The ball will appear to hang in space if enough of the in-betweens are drawn close to the topmost position.

4  Turn off Views > Show Ghosting and then play the animation. Concentrate on the movement of the ball. Adjust the curve handles some more while the animation plays. Observe the effect.

   The ball leaves the table as quickly as it hit it, then begins to slow down as it rises.

5 Play the animation, and then stop.

   The ball has bounce now. It looks like there is gravity at work here.

   When you see something you like in the viewport, it's a reminder that you should save your work. It's easy to forget while you're being creative.

6 Save your work as **mybounce.max**.

You've made the ball bounce once. In the next section, you'll learn to repeat the bouncing of the ball using Out-of-Range types in Track View.

Adding Parameter Curve Out-of-Range Types

You can repeat a series of keys over and over in a variety of ways, without having to make copies of them and position them along the time line. In this lesson, you'll add Parameter Curve Out-Of-Range Types to the ball's position keys. Out-Of-Range Types let you choose how you want the animation to repeat beyond the range of the current keys. They have the advantage that when you change one set of keys, the changes are reflected throughout the animation

Most of the tools in Track View are available both from menu choices and from toolbars. This feature is also on the Controllers menu.

Repeat keyframed motion:

1 Continue from the previous lesson, or open *bounce_repeat.max*. This is a 3ds Max scene with the ball bouncing once.

2 If the Curve Editor isn't displayed already, select the ball in any viewport, right-click, and choose Curve Editor from the quad menu.

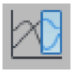
3 In the Controller window, make sure that only the Z Position track is selected.

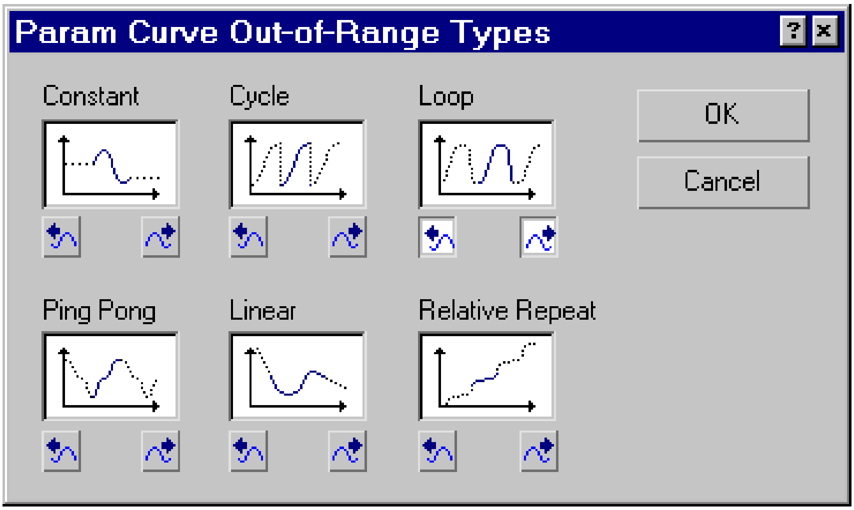
Before you repeat the keyframes, you'll extend the length of the animation.

4  Click Time Configuration. This button is beneath the Go To End button in the animation playback controls at the bottom-right corner of the interface (not Track View).

5 Change the Animation End Time to **120**.

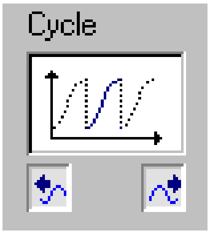
This adds 90 blank frames onto the existing 30 frames. It doesn't stretch the 30 frames over 120. The ball will still be bouncing once between frames 1 and 30.

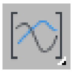
6  Now, back in Track View, click the Parameter Curve Out-Of-Range Types button on the toolbar.



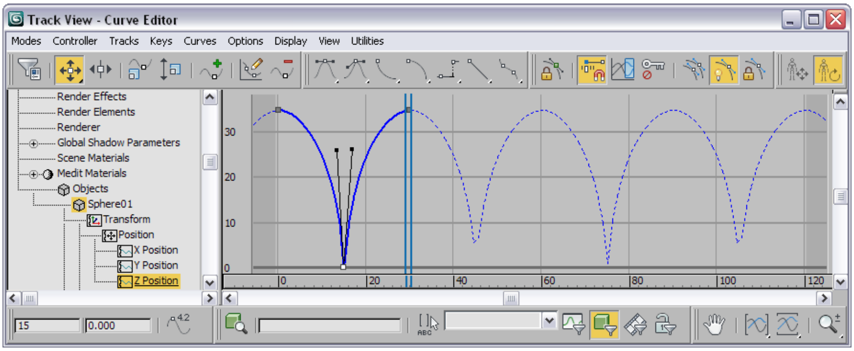
Parameter curve out-of-range type choices

7 Click both boxes under the Cycle graph to choose Cycle for In and Out. Click OK.



8  Click Zoom Horizontal Extents on the Navigation: Track View toolbar at the lower right corner of the Track View window.

The Keys window zooms back so the entire time segment is visible. The Parameter Out-of-Range curves are displayed as dotted lines.



Parameter Out-of-Range curve

There are no keys beyond frame 30. Any change made to the original keys will be reflected in the looping.

TIP You can create keys for the out-of-range curves by choosing Utilities > Track View Utilities > Create Out of Range Keys.

---

9 Play the animation.

The ball bounces over and over.

10 Save your work as **mybounce_repeat.max**

Next, you will learn to link the ball to a dummy object and then use the dummy to animate the position of the ball. This allows you to keep the bouncing independent from the position, and makes it easier to control the animation.

Animating with Dummy Objects

In this lesson, you will link the bouncing ball to a helper object. Then you can animate the helper so that the ball bounces across the top of some text. This animation technique is useful because you can control the ball's bouncing and its traveling motion independently.

Set up the lesson:

1  On the Quick Access toolbar, click the Open File button.

2 Navigate to the *\animation\auto_key* folder and open *bounce_dummy.max*.

This file is similar to the bouncing ball created in the last lesson. The only difference is that it has a text object prepared for you in the scene and it has a longer active time segment.

TIP If you want to keep using your own bouncing ball, you can merge the text object in from the *bounce_dummy.max* file by going to the Application menu and choosing Import > Merge.

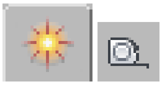
---

3  If you didn't open *bounce_dummy.max*., you'll need to extend the active time segment to 240 frames. Click the Time Configuration button and then in the Animation group change End Time to **240**.

Create a dummy object:

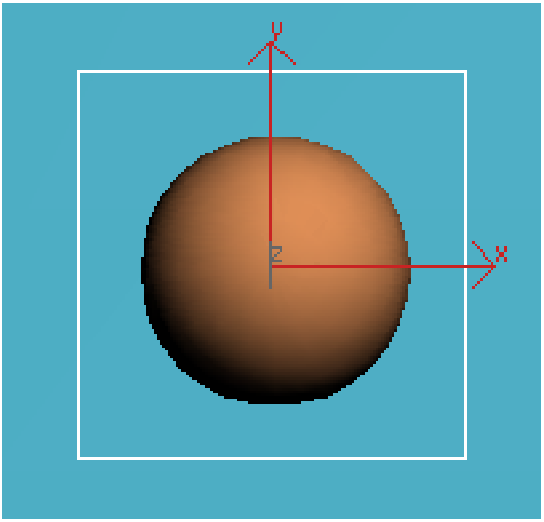
1 Go to frame 1.

2 Click the Top viewport to activate it, then zoom in on the ball and the box.

3  On the Create panel, click the helpers button, then on the Object Type rollout click Dummy.

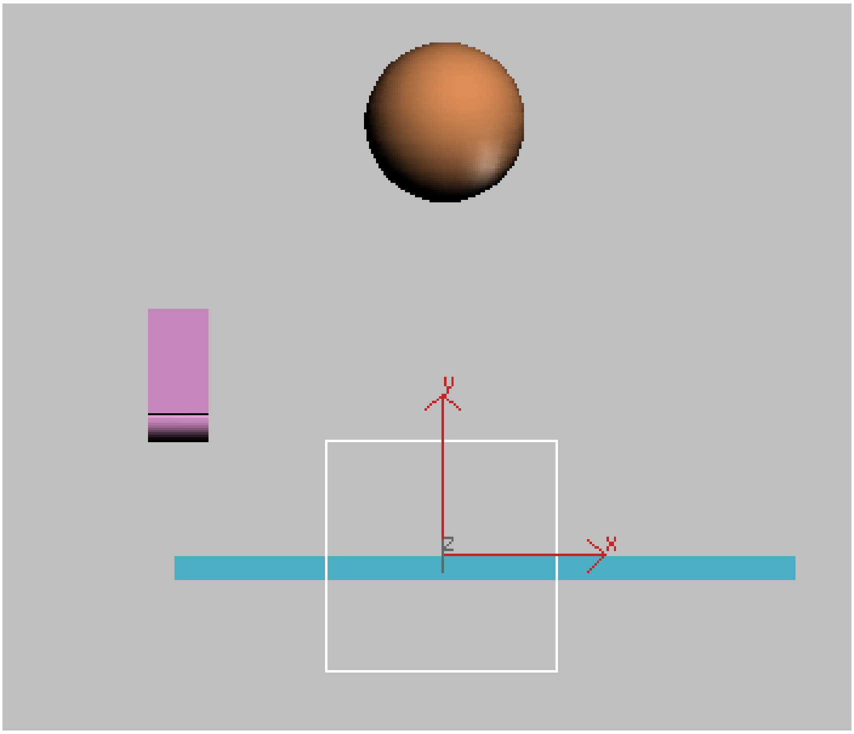
4 In the Top viewport, move the cursor over the ball.

5 Hold down the mouse button and drag outwards to create a dummy object.



Make the dummy in the Top viewport.

If you look in the Front viewport, you'll see that while the ball is up in the air, the dummy is positioned at the same level as the box.



Dummy created below the ball

Next you will align the dummy so that it is centered over the ball when viewed from the top.

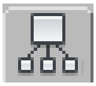
6  On the toolbar, click Align. Then in the Top viewport, click the ball.

The Align Selection dialog is displayed.

7 On the Align Selection dialog, turn on X Position and Y Position, but leave Z turned off. Click OK.

You'll see the dummy shift position so it is aligned to the ball.

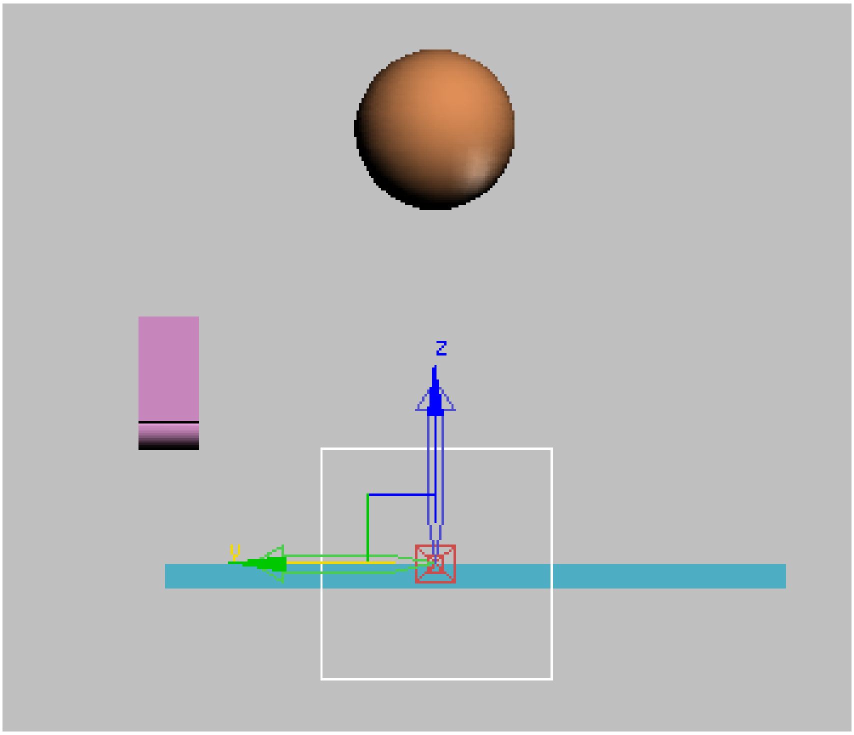
Next, you'll align the dummy's pivot point with its base, and you will position the dummy so that it sits on top of the box. The idea is to set up the dummy so that its pivot point will match where the ball bounces. Then placing the dummy on any frame in time will ensure the correct alignment.

8  Select the dummy object, and go to the Hierarchy panel.

Instead of moving the pivot, you will move the object using Affect Object Only. This moves the object but leaves the pivot unmoved.

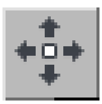
9 On the Adjust Pivot rollout, click Affect Object Only to turn it on.

The pivot icon is displayed in the viewport.



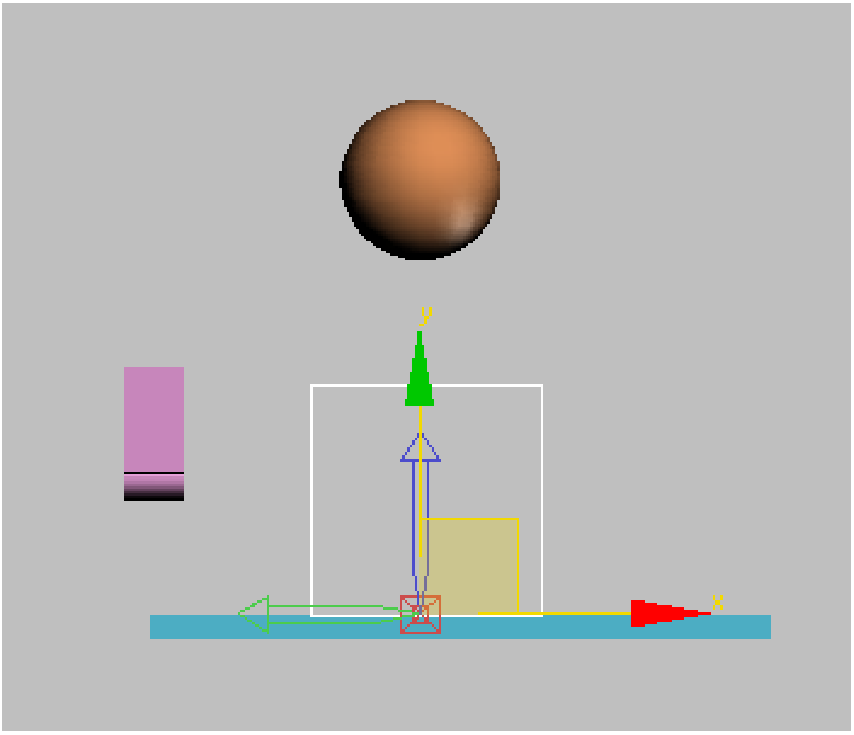
Pivot point tripod display

Now you can move the object to change its relationship with the pivot point.

10  Turn on Select And Move, and raise the dummy cube so its base is level with the pivot tripod. Use the Transform gizmo so the dummy only moves up.

You don't want to move in any other axis, since the dummy is already centered properly in the other axes.

TIP You can increase the size of the transform gizmo by pressing the = key repeatedly until the gizmo is the size you want.



= key used to grow the transform gizmo

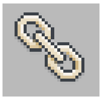
---

11 Turn off Affect Object Only.

Now you will link the ball to the dummy. The dummy will become the parent to the bouncing ball.

Link the ball to the dummy:

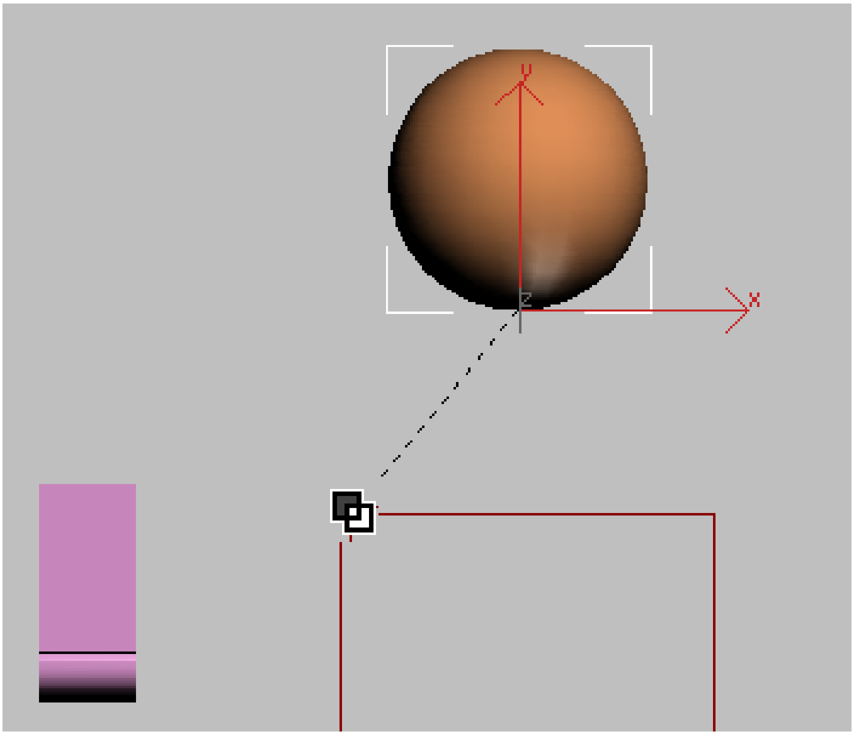
1 In the Front viewport, zoom in so you can see the dummy and the ball.

2  On the main toolbar, turn on Select And Link.

3 Move the cursor over the ball, then press and hold the mouse button.

The cursor changes to two interlinked boxes.

4 Move the mouse to the dummy. A rubber-band line follows the cursor. When the cursor passes over the dummy, it changes again. One box is white, showing you this object (the dummy) will be the parent of the first object (the ball). When the cursor has changed, release the mouse button.



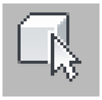
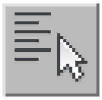
Link the ball to the dummy

You just linked the ball to the dummy.

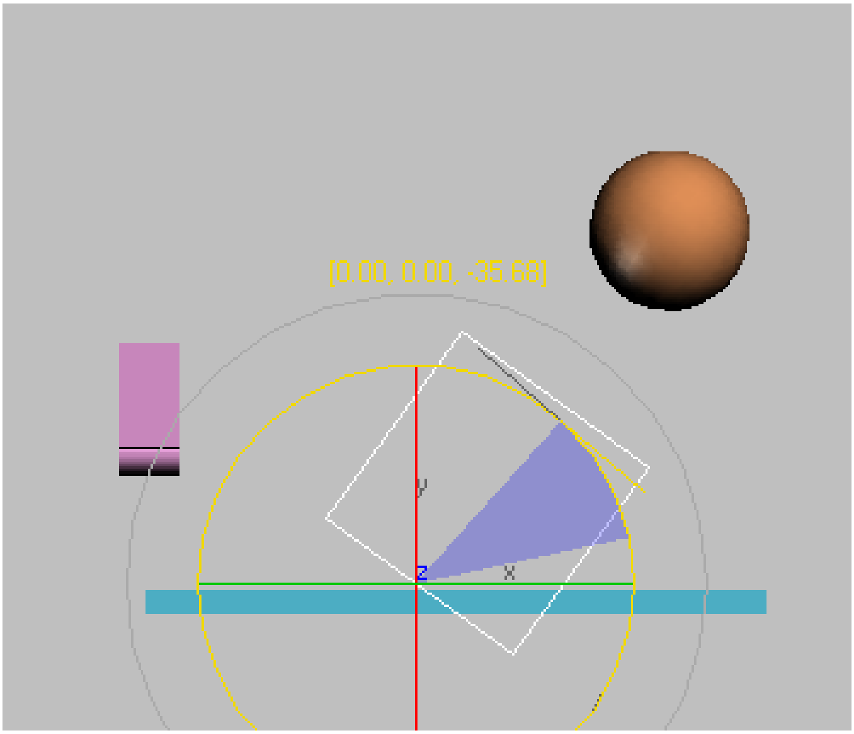
You can also create linkages in Schematic View. For something this simple, it's easier to link directly in the viewport.

When you've created a linkage, it's a good idea to test it out to make sure that you did what you think you did.

Verify that you've created the hierarchy:

1.  On the toolbar, turn on Select Object.
2.  Press H to open the Select From Scene dialog. Choose Display > Display Children to make sure this toggle is on, then choose Display > Expand All.

   The *Sphere01* object should appear indented below *Dummy01* in the object list. (You might have to click the plus (+) icon next to Scene Root in order to see the full list.)
3. Test the linkage by transforming the parent object. Rotate the dummy in the viewport. The ball should rotate as well.
4. Undo the transform after you've tested your linkage.



Testing the linkage

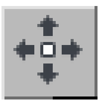
Now you're ready to animate the dummy. You'll use simple Auto Key animation first, so you can understand the process.

Animate the dummy:

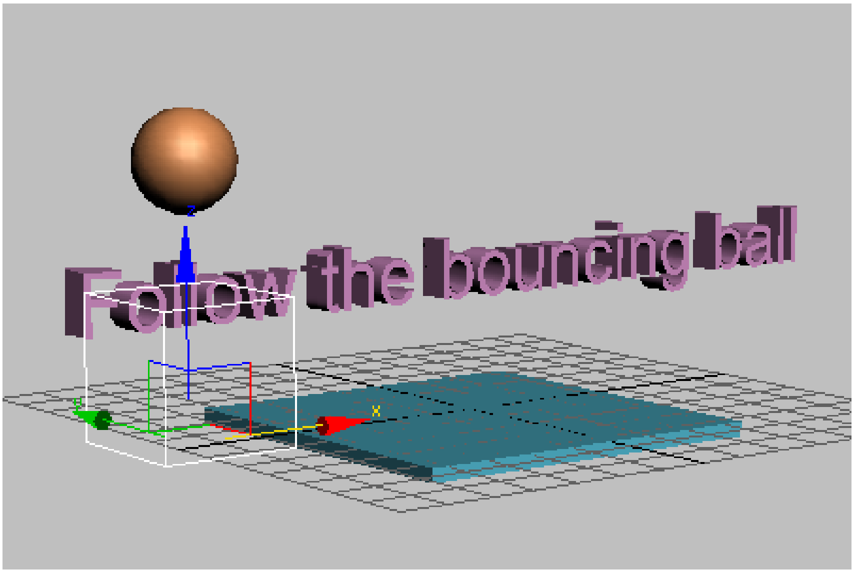
1 Grab the divider lines between the viewports and drag them so the perspective viewport is wide screen.

2 

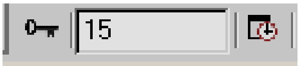
Turn on Auto Key

3  On the main toolbar, turn on Select And Move.

4 At frame 1, move the dummy so it is to the left of the box in the Perspective viewport.

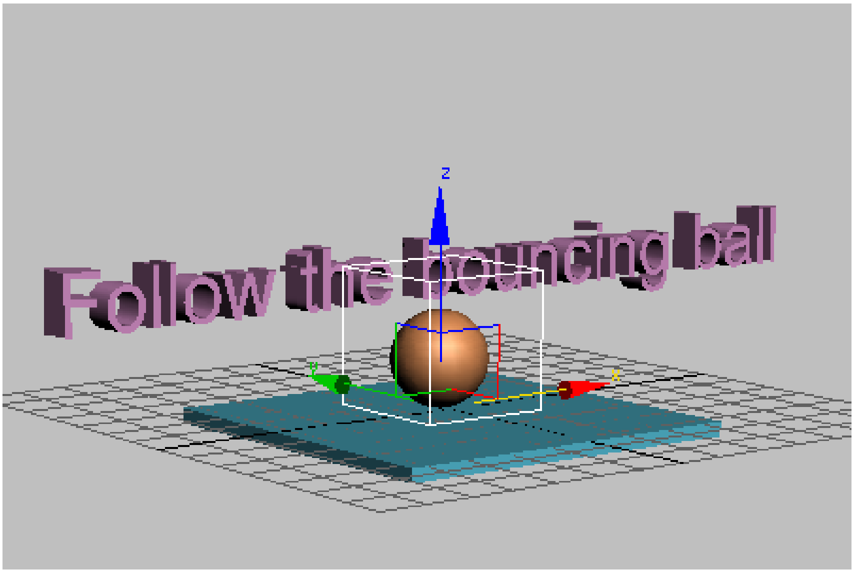


Dummy at frame 1

5 

Use the time slider to move to frame 15, or type **15** in the Go To Frame field.

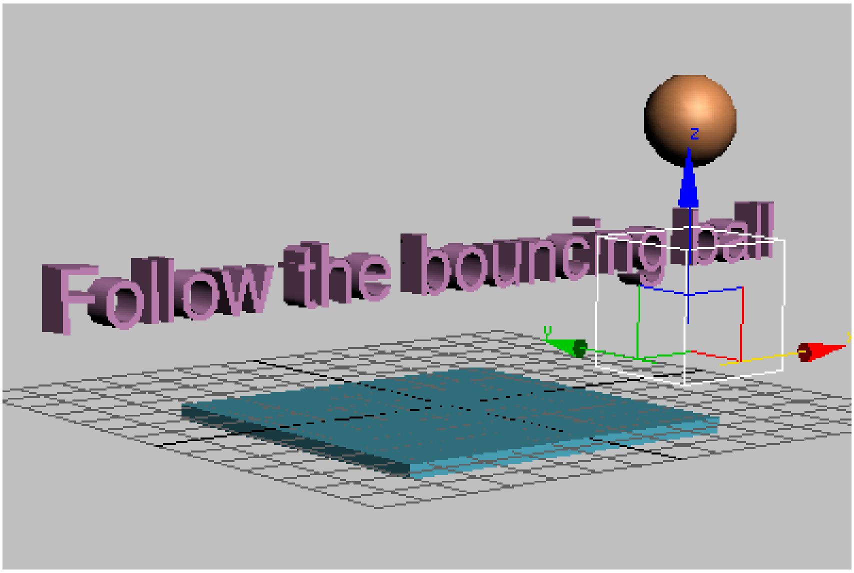
6 Move the dummy using the Transform gizmo so the ball is touching the box.



Dummy at frame 15

You just set two keys for the dummy, one at the start location at frame 0 and a second at frame 15.

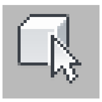
7 Go to frame 30 and move the dummy again to the right of the box, so the ball continues to bounce away, rather than straight up in the air.



Dummy at frame 30

You've now set a third key at frame 30. If you play the animation, you'll see the ball bounce off the box as if it had been tossed.

You can display the trajectory of the ball to help visualize the animation. Here's how.

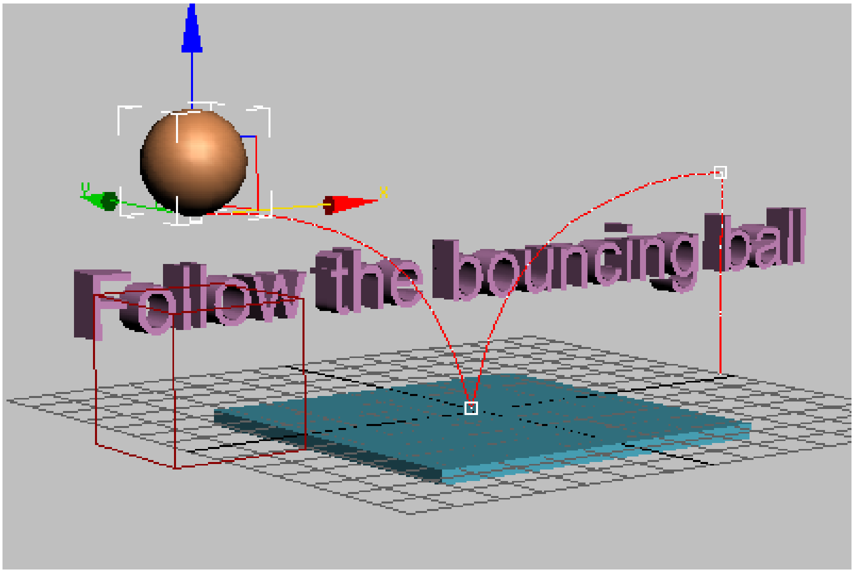
8  On the main toolbar, turn on Select Object.

You can use any transform tool for this, but using Select Object ensures that you don't accidentally transform the ball.

9 Select the ball, and then right-click it.

10 Choose Object Properties from the quad menu.

11 In the Object Properties dialog, in the Display Properties group, turn on Trajectory.



Trajectory display for the ball

12 Play the animation.

You see the ball bouncing onto the box and off, following the trajectory.

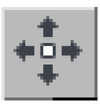
Try Layout mode:

If you turn off Auto Key and move the dummy, you will be moving the entire animation in space. When both Auto Key and Set Key are off, you are working in what is known as Layout mode. Here you will use Layout mode, so that instead of the ball bouncing on the block, the ball bounces off the letter F.

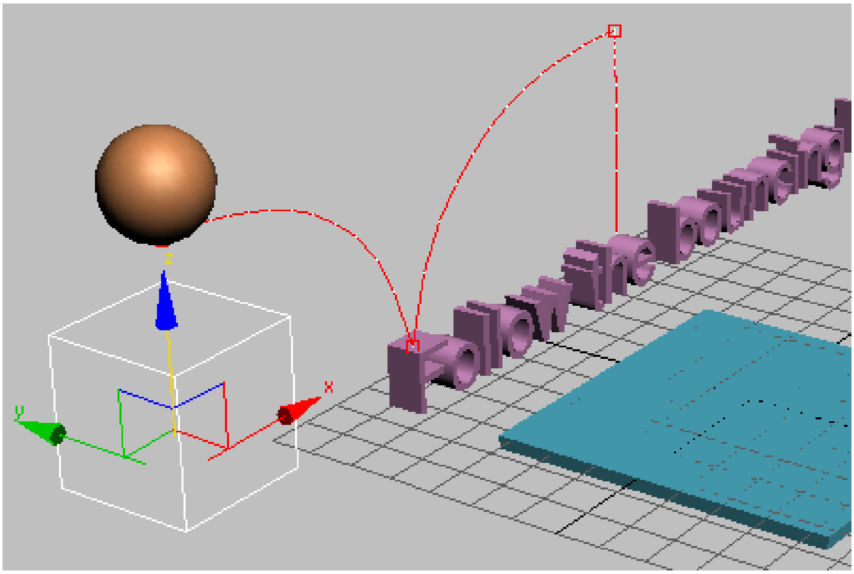
1 

Turn off Auto Key Mode.

The red disappears in the time slider background and viewport outline.

2  Move the dummy object back toward the text.

3 Watch the position of the trajectory and move the dummy until the bounce point of the trajectory intersects the top of the letter F.



Layout Mode lets you move the animation in space.

Bounce the ball on the letters:

Now you'll repeat what you've learned to create the ball bouncing on the letters.

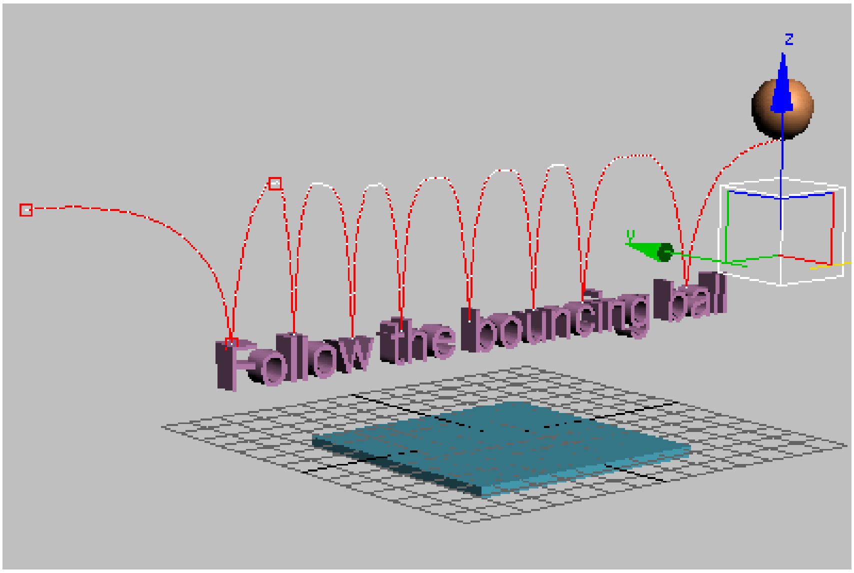
The ball bounces 8 times, making contact with the letters at frames 15, 45, 75, 105, 135, 165, 195, and 220.

1. 

Turn on Auto Key.
2. In the track bar, select the key at frame 30 and delete it.
3. Move the time slider to frame 45 (or enter **45** in the Current Frame field).
4. Position the dummy so the ball bounces on the double L's in the word "Follow".

5 Move the time slider, then the dummy so the ball bounces on the following letters at these frames.

- **F** at frame 15
- **ll** at frame 45
- **w** at frame 75
- **th** at frame 105
- **b** at frame 135
- **u** at frame 165
- **c** at frame 195
- **ba** at frame 225



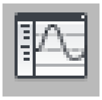
6 At frame 240, move the dummy so the ball move away from the letters.

7 Play the animation and observe the results.

8 Save your work as **mybounce_text.max**. If you had any trouble, you can open the file *bounce_text.max* to see the correct animation so far.

Next you will learn to use a multiplier curve to affect the height of the bouncing ball.

Add a multiplier curve:

1  Select the ball in the viewport, right-click and choose Curve Editor.

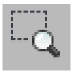
   The Curve Editor window is displayed, if it wasn't already visible.

2 In the Controller window, click the Z Position track.

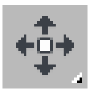
3 On the Curves menu, choose Apply Multiplier Curve. In the Controller window, click the plus icon. Click the Multiplier Curve to highlight it, and then hold down Ctrl and click the Z Position track. This way you have only these two curves displayed.

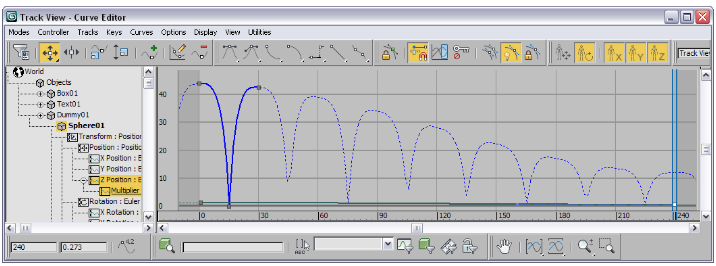
   In the Controller window, the Multiplier curve is added beneath the Z position track. It's not very noticeable in the Keys window on the right.

   The scaling of the multiplier track is quite small, so the slightest change to a key could result in a large change in the animation. You can counteract this by zooming in on the multiplier track.

4  On the Navigation: Track View toolbar (at the lower right of the Keys window), click the Zoom Region button. Drag a zoom region window around the key at frame 240 on the multiplier track.

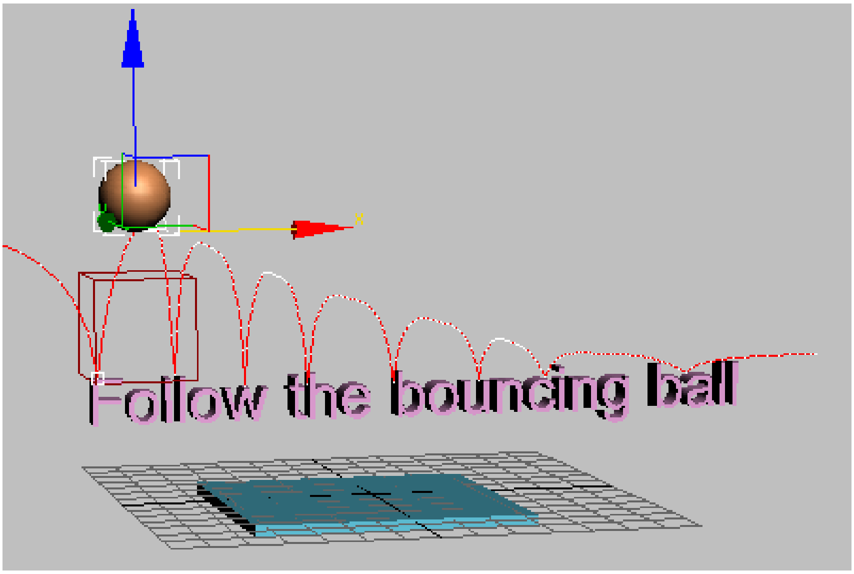
5 On the Options menu, turn on Interactive Update.

6  On the Track View toolbar click Move Keys to turn it on, then move the multiplier key downward, while observing the effect on the trajectory in the Perspective viewport.



Lowering the multiplier curve shows effect on Z position curve in Keys window

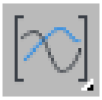
Don't move it below the horizontal zero value, or you will get some strange effects.



Multiplier curve shown on trajectory

TIP You can type in precision values on the Key Stats: Track View toolbar.

While working with Multiplier curves, if you're not sure you like the results, you can just turn them off. Select the Multiplier curve in the Controller window, then on the Curves menu choose On/Off.

7  Click Zoom Horizontal Extents on in the Navigation toolbar to see the entire curve again.

Using the Dope Sheet Editor

Track View also has a mode called Dope Sheet, which lets you work with keys and ranges. In this lesson, you'll use the range function to make your animation go faster. You'll also use the Time tools to reverse your animation.

Speed up the animation:

The bouncing ball doesn't have enough pep. To speed up the animation, you'll use Edit Ranges in Dope Sheet mode.

1 Continue from before or open *bounce_multiplied.max*.

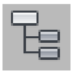
2 Select the dummy object in the viewport. Then, on the Graph Editors menu, choose Track View — Dope Sheet.

3  On the Keys: Dope Sheet toolbar, click Edit Ranges. By default, the Keys: Dope Sheet toolbar is on the top left.

The Keys window now displays the ranges for the animation.

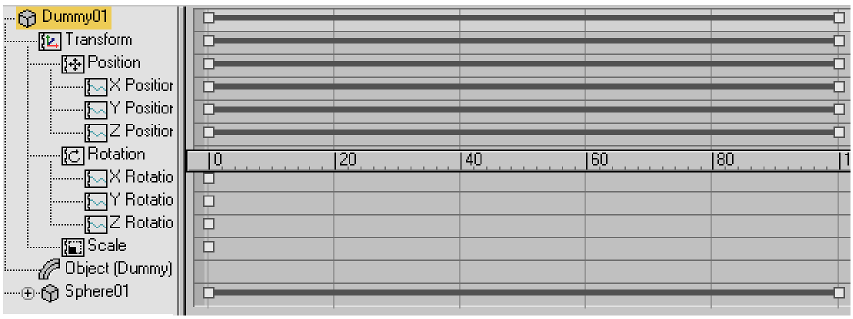
4 In the Controller window, highlight the item label *Dummy01*. This way you will adjust the ranges of all the dummy's tracks at the same time.

Before making changes to the dummy, you want to make sure that you also are making changes to the bouncing ball as well. Since the bouncing ball is the child of the dummy, you'll use the Modify Child Keys button.

5  On the Display: Dope Sheet toolbar, click Modify Child Keys to turn it on.

Now the changes you make to the dummy range will also be applied to the bouncing ball.

6 Click the end of the Dummy range and drag it to the left to around frame 100.



Range bars used to speed up the animation

This compresses the animation for the dummy and the bouncing ball so it happens within 100 frames.

TIP You can raise the time ruler up from the bottom of the Keys window for greater precision.

---

7 Play the animation.

The animation plays faster. The ball continues bouncing at the end of the animation. There are several different ways you could correct this. You could try to use an ease curve to stop the animation, or create keys from the out-of-range curve, and then delete the keys. Or you can set the active time segment to 100 frames.

8  Click the Time Configuration button beneath the animation playback controls.

9 Change the Animation End time to **100**.

Reverse time:

You can reverse the animation by using the Time tools available in Dope Sheet mode. It's easy to do.

1  On the Keys Dope Sheet toolbar, click Edit Keys.

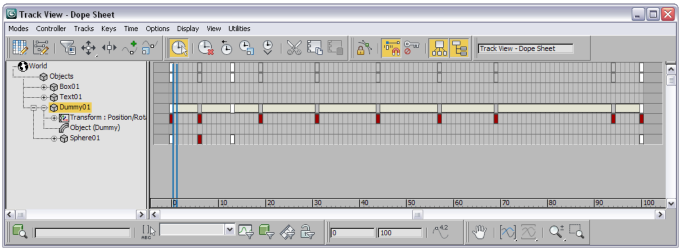
The range bars are replaced with keys.

2 On the Time menu, choose Select.

When working with Time commands, you first select the time, then make changes to it.

3 In the Keys window, on the Dummy track, drag from frame 0 to frame 100 to select the time.

The time is displayed as a light yellow band in the Dummy track.



Time displayed as light yellow band

4 On the Time menu, choose Reverse

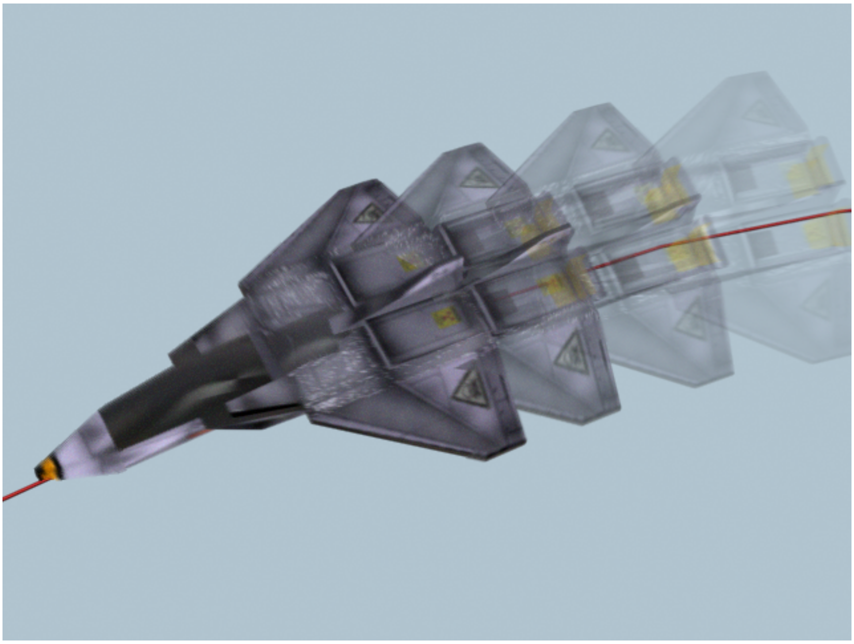
The animation plays backwards. The ball bounces from right to left instead of left to right, and the last bounce now happens on the letter F instead of the double L in ball.

TIP You can easily reverse the playback of an animation using controls found in the Time Configuration menu. But if you need to reverse the keys themselves, this is the technique to use.

Summary

These Animation tutorials introduced you to the techniques of creating animation. You learned how to animate a bouncing ball using the Auto Key button and transforms, how to control in-betweens using key interpolation and ghosting, and how to loop animation using Track View controls. Finally, you've learned about animating with dummy objects, using the Dope Sheet editor and weighted list controllers.

Flying a Spacefighter



In this tutorial, you'll animate a spacefighter to make it fly along a simple path by using the path constraint. You'll also be shown how to blend paths.

In this tutorial, you will learn how to:

- Animate with Path constraints.
- Use dummy objects for animation.
- Use weighted list controllers to add turbulence.
- Set keys using the Set Keys button.
- Control Euler controller rotations.
- Substitute high-resolution objects for low-resolution objects by using XRefs.

Skill level: Beginner to Intermediate

Time to complete: 1 hour

Adding and Adjusting Flight Paths

In this first exercise, you'll assign a path constraint to the spacefighter and have it fly along a path. You'll also set a few path parameters to improve the flight dynamics of the spacefighter.

Set up the lesson:

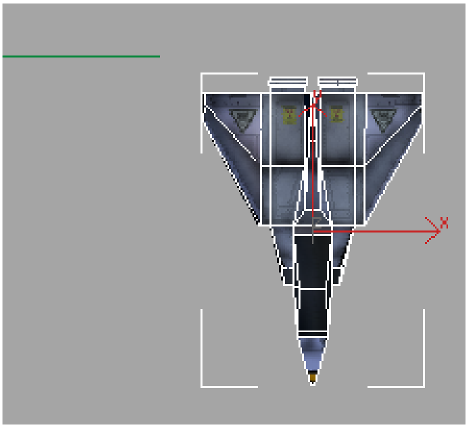
-  On the Quick Access toolbar, click the Open File button, navigate to the *animation\spacefighter* folder, and open *flyingspacefighter.max*.

This scene includes the following:

- A spaceship named *SpaceFighter*.
- A path, *Path01*.
- A (hidden) camera named *SpaceCam*.

Assign a path constraint:

1 In the Top viewport, select the *SpaceFighter* object.



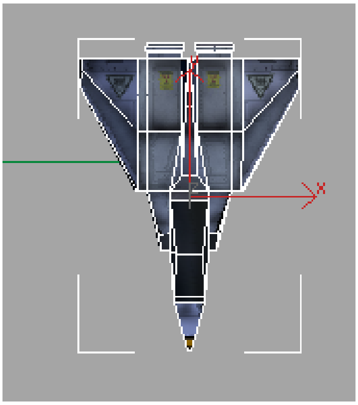
2 Open the Animation menu and choose Constraints > Path Constraint.

A dotted line now links the spacefighter and the mouse cursor.

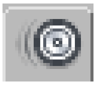
3 Click the green line, *Path01*.

Once you pick the path, the spacefighter jumps to the start point of the path.

(Any spline can become an animation path.)



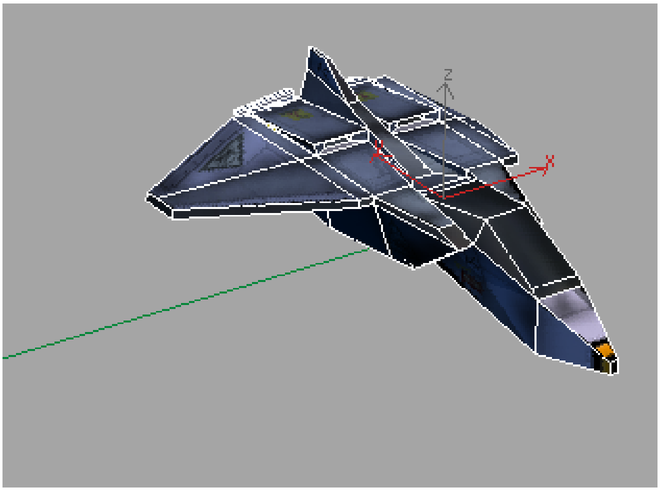
Also, the command panel automatically switches to the Motion panel.

4  From the Motion panel, slide the rollouts up until you can see all of the Path Parameters rollout, if necessary.

You'll see that *Path01* has been added to the path list.

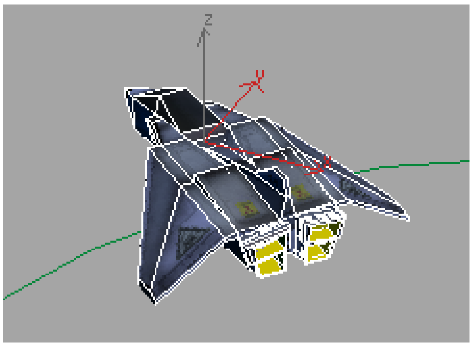
5 Activate the SpaceCam viewport and play the animation.

The spacefighter moves along the path, but it doesn't point in the correct direction.

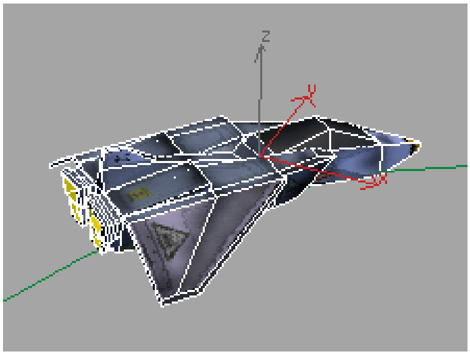


6 In the Path Options group of the Path Parameters rollout, set the following:

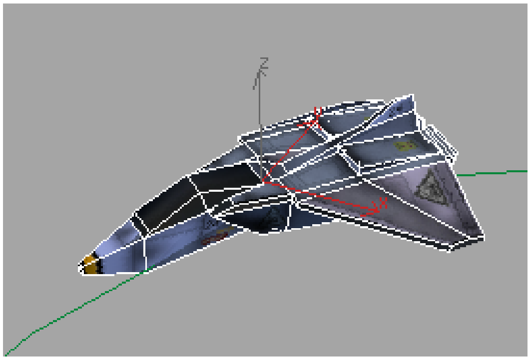
- Turn on Follow: the spacefighter follows the path and turns as the path curves, but it’s perpendicular to the motion path.



- In the Axis group, change the axis to Y: the spacefighter is reoriented and faces along the path, but it's flying backwards.



- Turn on Flip: the spacefighter now faces the direction it moves along the path.



NOTE You can change settings while the animation is playing.

7 Play the animation again.

Now the spacefighter is moving along the path properly, but its flight dynamics don't look realistic.

Improving the Flight Characteristics

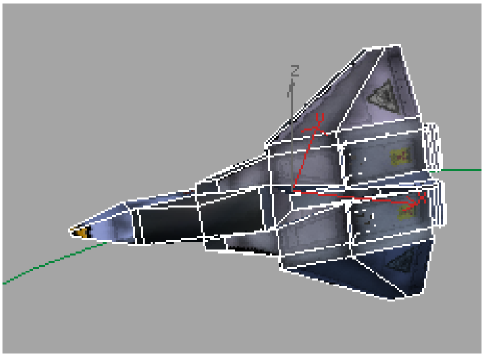
In this section, you'll improve the flight characteristics of the spacefighter. You'll make it move more realistically as it enters and exits turns.

Make the flight more realistic:

1 Turn on Bank and play the animation again, if it isn't still playing.

The spacefighter now banks as it goes through the turns. However, the motion is subtle; it needs to be more pronounced. You'll use the Bank Amount and Smoothness settings to make the spacefighter look as though it's banking sharply into the curves of the path.

2 Set Bank Amount to **7.0**.



Bank Amount controls how far an object rolls to either side of the path it travels along.

If you were sitting in the cockpit, positive values roll the starfighter to the left and negative values roll to the right. Be careful how high you set Bank Amount. If too high, the spacefighter will roll all the way over. Experiment with different settings and put it back to 7.0 before continuing.

3 Set Smoothness to **1.0**.

Smoothness controls how rapidly the roll changes as the starfighter moves through bends in the path. Smaller values make the object more responsive to changes in the curve while larger values smooth out jerkiness.

The motion of the spacefighter as it rights itself coming out of the turns is more even. Try increasing and decreasing the Smoothness value to see what happens.

4 Save your scene as **MySpaceFighter01.max**.

Animating the Path of the Spacefighter

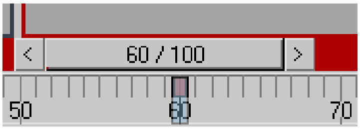
You can animate the Path Parameters settings for more dynamic-looking animation. In this section, you'll add a bit of drama to the action by animating some settings changes.

Animate path parameters:

1 Move the time slider to frame 60.

2 

Turn on Auto Key and set Bank Amount to **6.0**.



You'll see a new key added to the timeline at frame 60.

3 Slide the time slider to 75 and set Bank Amount to **12.0**.

4 Turn off Auto Key and play the animation.

As the spacefighter enters the second curve, it makes a drastic rolling turn, as if evading a missile or dodging a laser.

5 Save your scene as **MySpaceFighter02.max**.

Blending Paths

So far, you've worked with a single path. 3ds Max allows you to combine motion along multiple paths, which can result in some interesting effects.

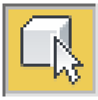
Set up the lesson:

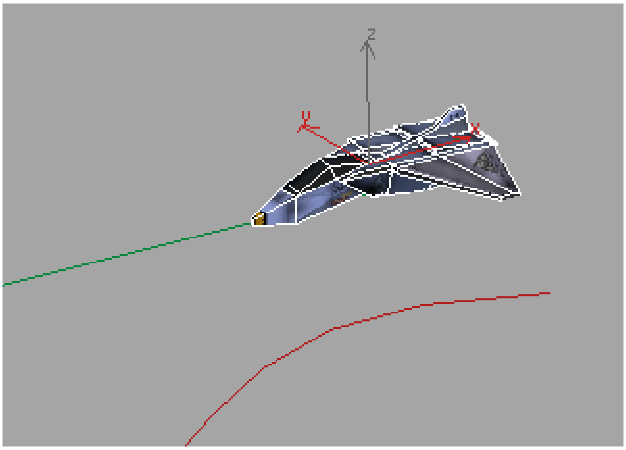
- Open the file *flyingspacefighter02.max*.
  Files for this lesson are in the *\scenes\animation\spacefighter* folder.

  This scene includes the following:

  - A spaceship named *SpaceFighter*.
  - Two paths, *Path01* (green) and *Path02* (red).
  - A camera (hidden) named *SpaceCam*.

Fly the spacefighter along blended paths:

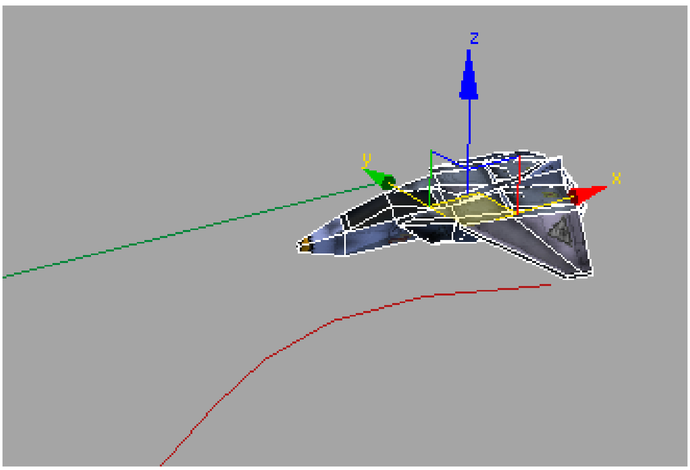
1  Select *SpaceFighter* and open the Motion panel.



The spacefighter is already constrained to *Path01*.

2 On the Path Parameters rollout, click the Add Path button.

3 Select *Path02*, the red path.



The spacefighter is now positioned halfway between the two paths. This is because each path is influencing the spacefighter equally.

4 Click the Add Path button again to turn it off.

5 Activate the SpaceCam viewport, if it's not already active, and play the animation.

The Weight setting controls how much the spacefighter is affected by each path.

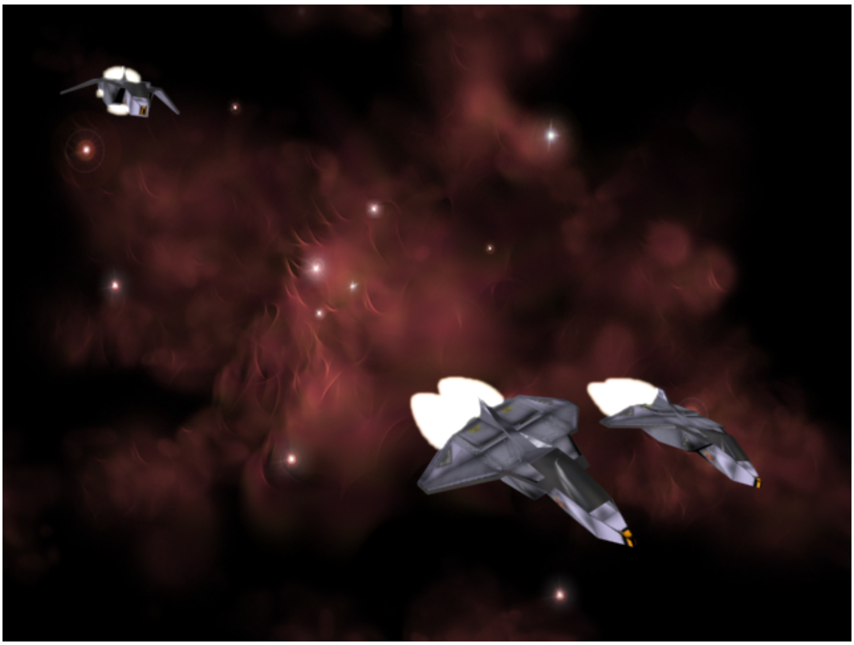
6 In the path list on the Path Parameters dialog, highlight the *Path01* entry and set its Weight value (immediately below the list) to **25.0**.

7 Play the animation again.

The spacefighter follows *Path02* more closely because it has a greater weight than *Path01*. Experiment with different Weight settings for each path and see what happens.

8 Save your scene as **MySpaceFighter03.max**.

Animating the Spacefighter with Constraints and Controllers



In this lesson, you'll use an assortment of constraints and controllers with a flight of spacefighter on a mission.

Set up the lesson:

- Open the file *flyingspacefighter03.max*.
  Files for this lesson are in the *\animation\spacefighter* folder.

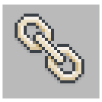
  The scene already contains the following:

  - A flight of three spacefighters, *FlightLeader, Wingman01,* and *Wingman02*.
  - A dummy object, *SpaceshipControl*.
  - Two motion paths, *flightpath (visible) & wingmanpath (hidden)*.
  - A camera (hidden), *SpaceCam*.

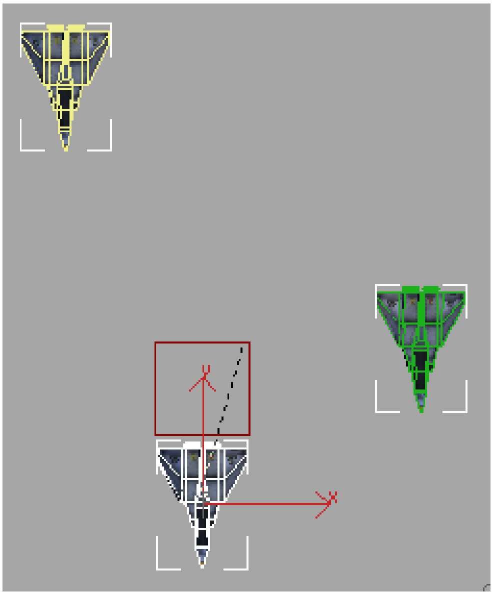
Using a Dummy Object to Control the Flight

Dummy objects are helper objects. You can create them from the Helpers button on the Create panel. They can be useful when setting up an animation. By animating a dummy, you can focus on getting your motion set up using a simple object that doesn't slow down your system. Once the dummy animation is complete you link objects to the dummy. Now wherever the dummy moves, the object goes with it.

Link the spacefighters to the dummy:

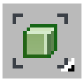
1.  Activate the Top viewport, if it's not already active, and click the Select And Link button from the Main toolbar.
2. Select *FlightLeader* and drag the cursor over the dummy, *SpaceshipControl*. Release the mouse button.

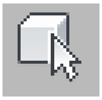
   FlightLeader is now linked to the *SpaceshipControl*. Wherever you move *SpaceshipControl*, *FlightLeader* will follow.



3 Link both *Wingman01* and *Wingman02* to *SpaceshipControl*.

Add a path constraint to the dummy:

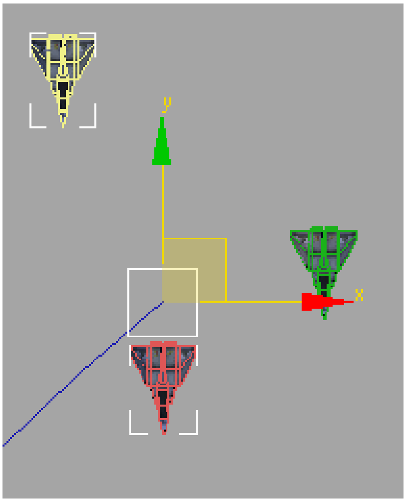
1  In the Top viewport, use Zoom Extents to view the scene.

2  Click Select Object to turn it on and turn off Select And Link.

3 Select *SpaceshipControl*.

4 From the Animation menu, choose Constraints > Path Constraint. Drag the cursor and click the *flightpath*.

   SpaceshipControl jumps to the beginning of the path. The spacefighters follow along since they're linked to the dummy.



5 In the Path Parameters rollout, make the following settings.

- Turn on Follow, Bank, and Constant Velocity.
- In the Axis group, turn on Y and Flip.
- Set the Bank Amount to **7.0** and the Smoothness to **1.0**.

6 Activate the SpaceCam viewport and play the animation.

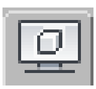
All three spacefighters move along the flightpath.

7 Save the scene as **MyFlight01.max**.

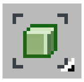
Making the Camera Follow the Action

In this exercise, you'll use the Link Constraint to make the camera follow the flight as it passes by. Continue from the last lesson or open *flyingspacefighter04.max*

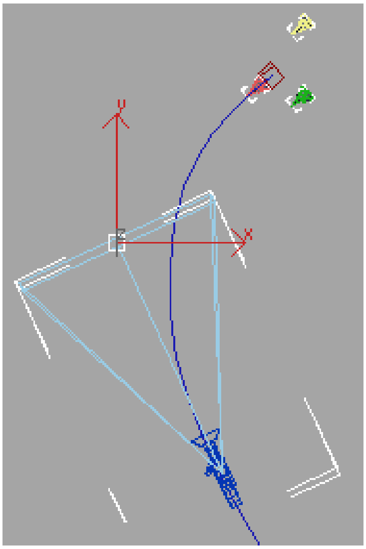
Add link constraint to the camera:

1  Open the Display panel and turn off Cameras in the Hide By Category rollout.

The camera, *SpaceCam*, will appear.

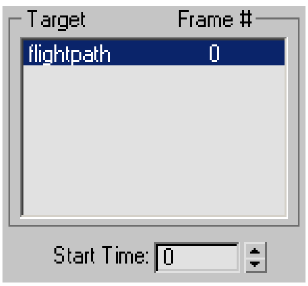
2  In the Top viewport, use Zoom Extents to view the scene. Select the camera target.

Also make sure the time slider is set to frame 0 (zero).

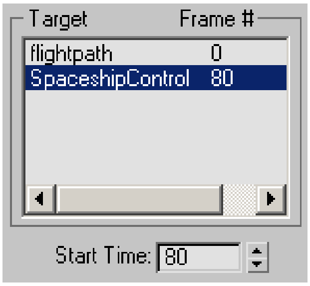


3 From the Main menu, choose Animation > Constraints > Link Constraint. Click the *flightpath* to set the first Link Parameter at frame 0 (zero).

NOTE For this step and the one that follows, Auto Key doesn't need to be turned on, because the Link constraint is active.



4 Move the time slider to frame 80 and click the Add Link button in the Link Params rollout. Click *SpaceshipControl*.



5 Click Add Link to turn it off.

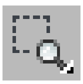
6 Activate the SpaceCam viewport and play the animation.

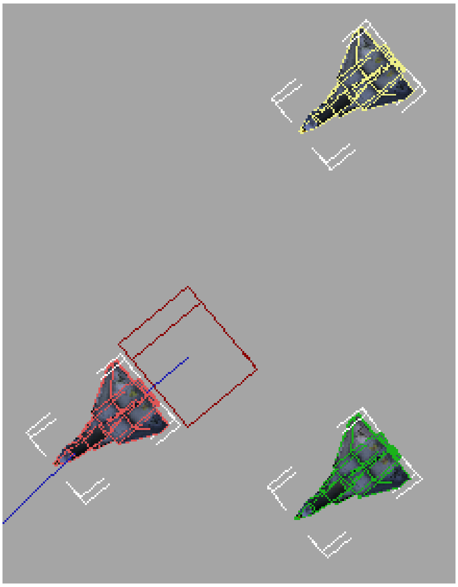
The camera target is linked to the *flightpath* from frames 0 to 79. At frame 80, the camera target begins to specifically track the SpaceshipControl object.

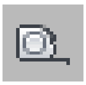
7 Save the scene as **MyFlight02.max**.

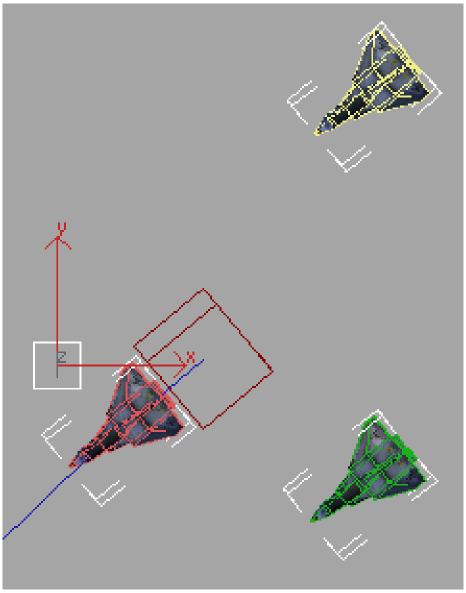
Your Flight Leader Gets Cocky

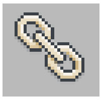
Your flight leader is a pretty bold character. He or she feels a victory roll is in order as the flight passes by your vantage point from *SpaceCam*. In this exercise, you'll use the Orientation Constraint to control the rotation of the flight leader's spacefighter as it performs a barrel roll. Continue from the last lesson or open *flyingspacefighter05.max*.

1  In the Top viewport, use Region Zoom to view the three spacefighters.

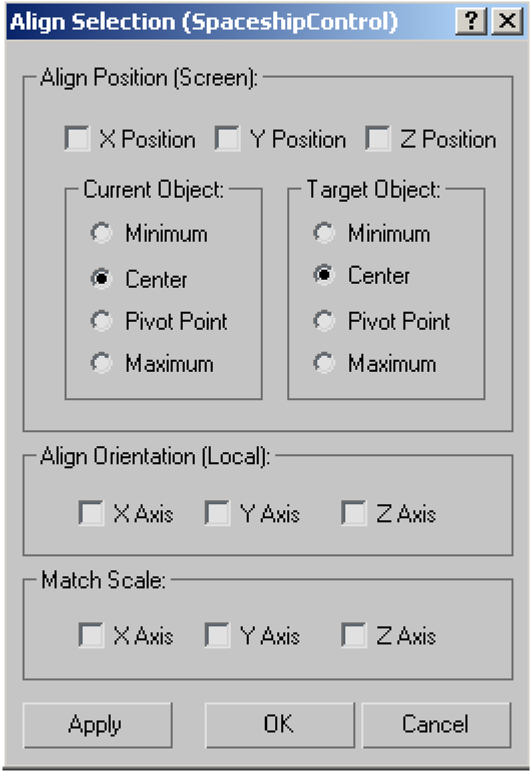


2  On the Create panel, click Helpers, then Dummy. Create a new dummy object near the *FlightLeader* and name it **barrelroll**.



3  Click Select And Link and link *barrelroll* to *SpaceshipControl*.

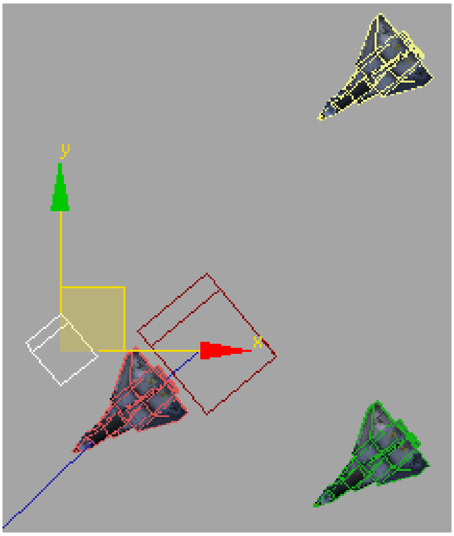
4 From the Tools menu, choose Align > Align, and click *SpaceshipControl*.
The Align Selection (SpaceshipControl) dialog appears.

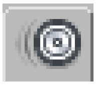


5 In the Align Position (Screen) group, turn off the X, Y, and Z Positions.

6 In the Align Orientation (Local) group, turn on the X, Y, and Z Axis controls and click OK.

The *barrelroll* dummy now has the same orientation as the *SpaceshipControl*.



7  On the Motion panel, open the Assign Controller rollout, if it's not open, and select the Rotation: Euler XYZ controller.

8  Click the Assign Controller button and choose TCB Rotation from the Assign Rotation Controller dialog.

NOTE Do not miss this step. It allows you to rotate the *barrelroll* dummy about its local Y axis.

Animate the victory roll:

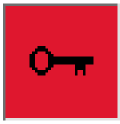
1 Select the *FlightLeader* in the Top viewport.

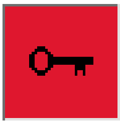
2 Open the Animation menu and choose Constraints > Orientation Constraint. Move the cursor over the *barrelroll* dummy and select it.

You'll see *barrelroll* added to the Orientation Constraint Target list on the Motion panel.

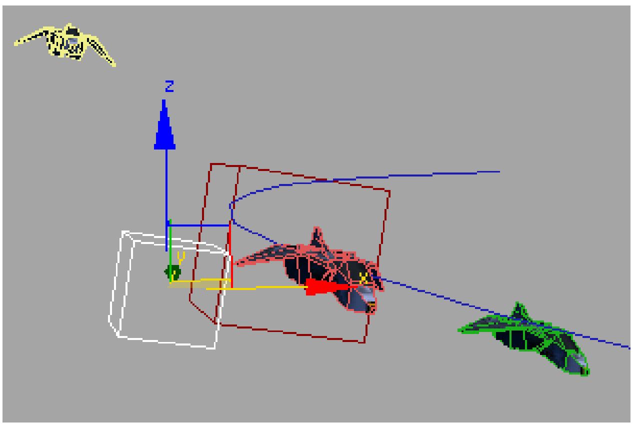
3 Select *barrelroll* and make sure that the time slider is at frame 0 (zero).

4 

 Turn on the Set Key toggle and click the Set Keys button.

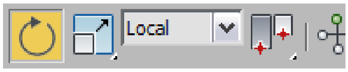
5  Move the time slider to frame 110 and click Set Keys again.

You've added two keys that will keep the *FlightLeader* flying normally from frames 0 to 110.



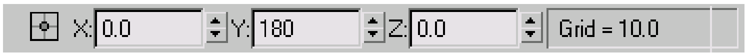
6 Activate the SpaceCam viewport and move the time slider to frame 130.

7  Click Select And Rotate and change the Reference Coordinate System to Local.



8 Click the *barrelroll* dummy and rotate it around the Y axis to about **180** degrees and click the Set Keys button.

NOTE If you'd like to be precise, you can enter the rotation in the Y axis type-in field on the status bar.



---

9 Move the time slider to frame 150 and rotate the *barrelroll* dummy around the Y axis another **180** degrees and click the Set Keys button. Turn off the Set Key Toggle when you've finished.

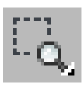
10 Save the scene as **MyFlight03.max** and play the animation.

A Wingman in Trouble

While the *FlightLeader* is performing stunts, *Wingman01* seems to be having some trouble. He doesn't seem to be flying as smoothly as the others. In this exercise, you'll use the Noise Controller to add some turbulence to *Wingman01* flight dynamics. Continue from the last lesson or open *flyingspacefighter06.max*.

NOTE If you continue from the last lesson, make sure that the time slider is back on frame 0 (zero).

Add turbulence:

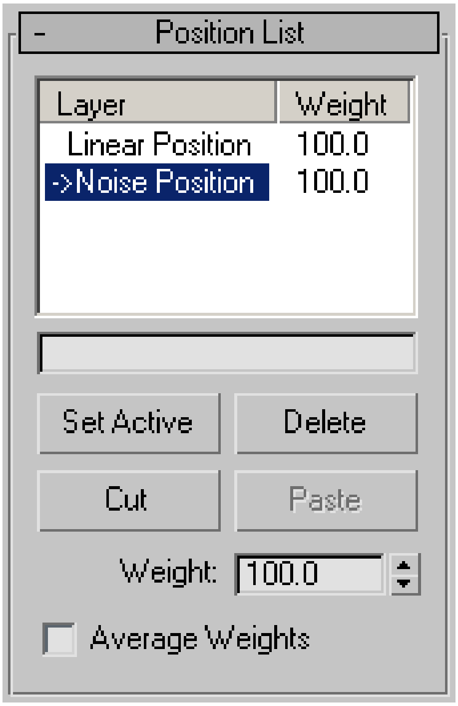
1  In the Top viewport, use Region Zoom to view the three spacefighters, if you haven't done so already.

You might have to do a Zoom Extents first, then a Region Zoom to see the three fighters.

2 Select *Wingman01*.

3 On the Animation menu, choose Position Controllers > Noise.

Doing this automatically adds a List Controller to the *Wingman01*. The Position List contains the original Linear Position and the new Noise Position controller with default Weight settings of 100.0 percent.



4 Play the animation and note the erratic flying of *Wingman01*.

5 Stop the playback and change the Weight of the Noise Position controller to **25.0** percent.

Now the flight path of the *Wingman01* spacefighter is affected by slight battle damage.

6 Save the scene as **MyFlight04.max**.

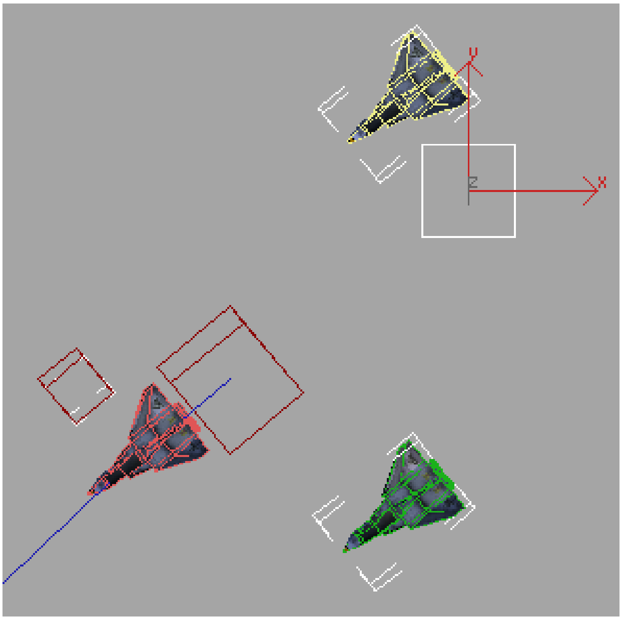
A Wingman Is Called Away

Now it looks like *Wingman02* has received a transmission and is being ordered to peel off and fly somewhere else. You'll revisit the Link Constraint to make *Wingman02* follow the *FlightLeader* for a little while then take off on another path. Continue from the last lesson, or open *flyingspacefighter07.max*.

Alter course for Wingman02:

1 Reset your time slider to frame 0 (zero) and zoom to the trio of spacefighters in the Top viewport.

2 Create a dummy near *Wingman02* and call it *WingmanControl*.

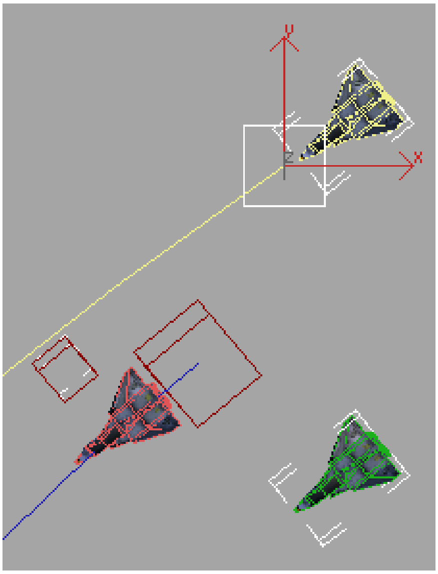


3 Right-click any viewport and choose Unhide By Name from the quad menu. Select *wingmanpath* and click Unhide.

A yellow path appears in front of *Wingman02*.

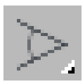
4 Make sure the new dummy, *WingmanControl*, is still selected and choose Animation > Constraints > Path Constraint. Drag the cursor over and pick *wingmanpath*.

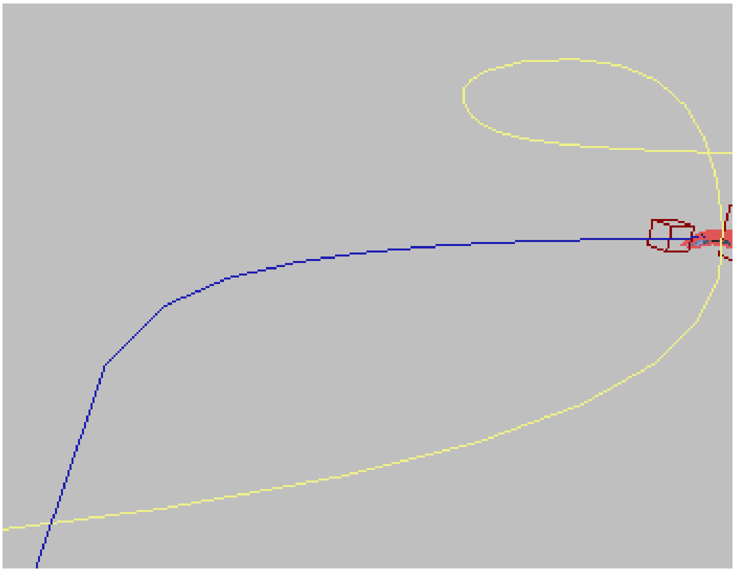
The *WingmanControl* jumps to the beginning of *wingmanpath*.



5 In the Path Parameters rollout duplicate the previous settings.

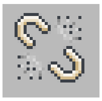
- Turn on Follow, Bank, and Constant Velocity.
- In the Axis group, turn on Y and Flip.
- Set the Bank Amount to **7.0** and the Smoothness to **1.0**.
  If you play the animation, you'll see *WingmanControl* on its own path.

TIP  To better see *Wingman02* fly away, before you play the animation go to frame 0, activate the SpaceCam viewport, turn on the Field-of-View navigation button, and drag downward in the SpaceCam viewport until in frame 0 you can see the entire loop of *wingmanpath,* and a bit of *FlightLeader,* at the right-hand side of the viewport.



SpaceCam viewport with a larger Field-of-View

Make Wingman02 change paths:

1.  In the Top viewport, select *Wingman02* and click the Select And Unlink button.

   Wingman02 is no longer linked to the *SpaceshipControl* dummy object.

2. From the Animation menu, choose Constraint > Link Constraint and select *SpaceshipControl*.

   Wingman02 will work as it did before, but the link constraint will give you the flexibility to have it follow a different path.

3. Move the time slider to frame 45 and click the Add Link button in the Link Params rollout on the Motion panel.

4. Click the *WingmanControl* dummy.

   You will see *WingmanControl* is added to the Target list, and when *Wingman02* gets to frame 45, the spacefighter begins to follow the *WingmanControl* dummy on the other path.

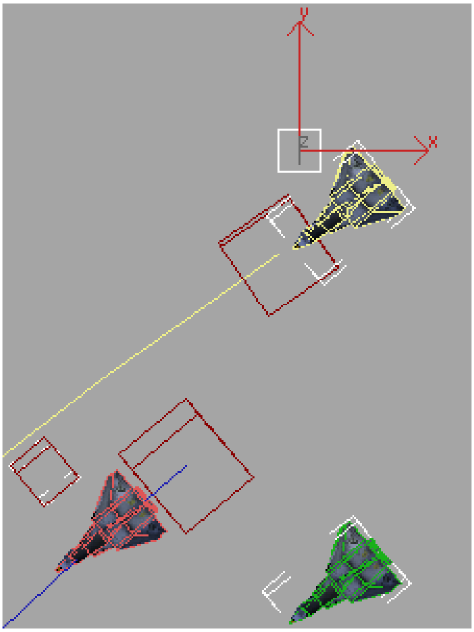
5 Save the scene as **MyFlight05.max** and play the animation.

Make Wingman02 roll out of formation:

To make *Wingman02* roll out of formation, you'll use the Orientation Constraint again.

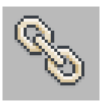
1 Move the time slider back to frame 0 (zero), if it's not there already.

2 Create another dummy object in the Top viewport near *Wingman02* and name it *wingmanroll*.



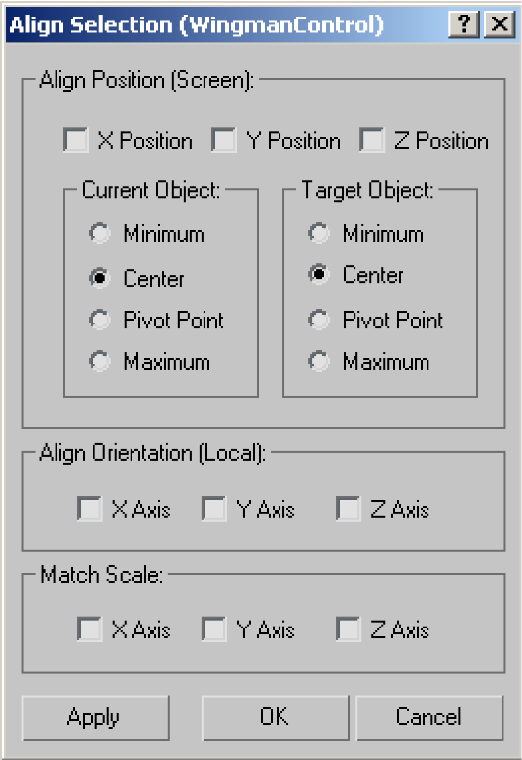
3 Use Select And Move to position *wingmanroll* next to *Wingman02*.

Watch the Front and Right viewports to help you position it. This will help you keep the objects in your scene organized.

4  Click Select And Link and link *wingmanroll* to *WingmanControl*.

5 From the Tools menu, choose Align > Align, and click *WingmanControl*.

The Align Selection (WingmanControl) dialog appears.

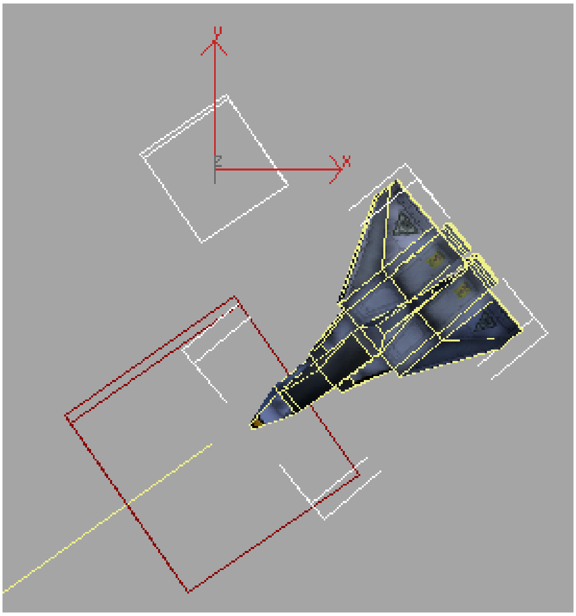


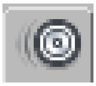
6 In the Align Position (Screen) group, make sure the X, Y, and Z Position controls are turned off.

7 In the Align Orientation (Local) group, turn on the X, Y, and Z axis controls and click OK.

The *wingmanroll* dummy aligns to *WingmanControl*.

NOTE This is important because you want to make sure that rotation values you give *wingmanroll*, later on, will be based on the initial orientation of *WingmanControl*. Otherwise, any rotation you give *wingmanroll* will result in the spaceship tumbling out of control.



8  On the Motion panel, open the Assign Controller rollout, and select the Rotation: Euler XYZ controller.

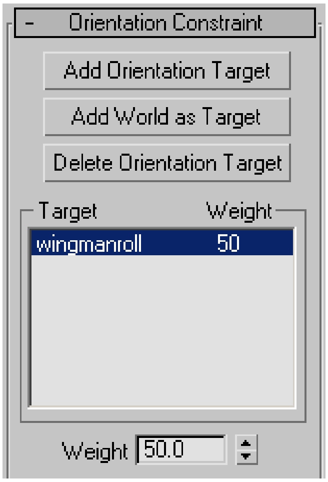
9  Click the Assign Controller button and choose TCB Rotation from the Assign Rotation Controller dialog.

NOTE If you don't assign the TCB Rotation controller, you will not be able to rotate *wingmanroll* about a local axis.

10 Select *Wingman02* in the Top viewport.

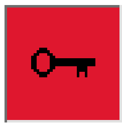
11 Open the Animation menu and choose Constraints > Orientation Constraint. Move the cursor over *wingmanroll* and select it.

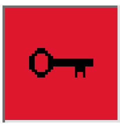
You'll see *wingmanroll* added to the Orientation Constraint target list in the Orientation Constraint rollout on the Motion panel.



12 Select *wingmanroll* in the Top viewport.

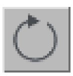
13 

 Turn on the Set Key toggle and click the Set Keys button.

14  Move the time slider to frame 60 and click Set Keys again.

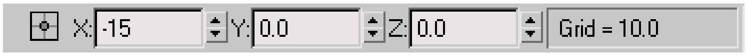
You've added two keys that will keep *Wingman02* flying normally from frames 0 to 60.

15 Activate the SpaceCam viewport and move the time slider to frame 85.

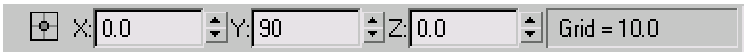
16  Click Select And Rotate and change the Reference Coordinate System to Local.

17 You'll make two rotations during this step:

- Enter **–15** in the X axis Coordinate Display Type-in field and click the Set Keys button.



- Enter **90** in the Y axis Coordinate Display Type-in field and click the Set Keys button.



18 Move the time slider to frame 100 to make the next two rotations:

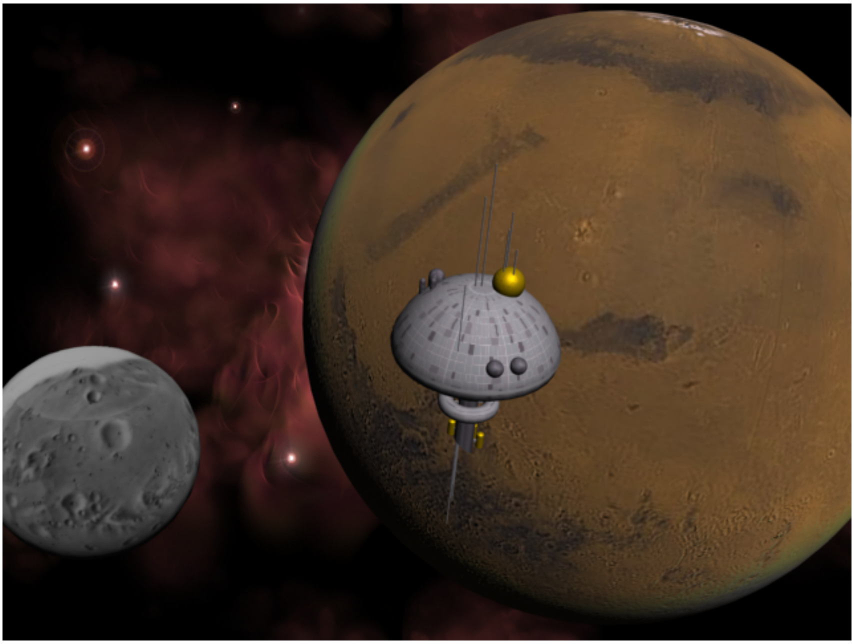
- Rotate **5** degrees around the Z axis and click the Set Keys button.
- Rotate **90** degrees around the Y axis and click the Set Keys button again.

19 Move the time slider to frame 115 to make the next two rotations:

- Rotate **10** degrees around the Z axis and click the Set Keys button.
- Rotate **90** degrees around the Y axis and click the Set Keys button again.

20 Turn off the Set Key toggle when you've finished. Save the scene as **MyFlight06.max** and play the animation.

Taking Control of Mars, Its Moons, and the Space Station



So far, you've used dummy objects to help animate the spacefighters. Another handy use of dummy objects is as an alternate pivot point. Any object can be used as a pivot, but dummies are great because they don't render.

Set up the lesson:

- Open the *spacestation.max* file.
  Files for this lesson are in the *\animation\spacefighter* folder.

  This scene includes the following:

  - Three heavenly bodies, *Mars* and its two moons, *Deimos* and *Phobos*
  - A space station named *SpaceStation*
  - A camera (hidden) named *SpaceCam*

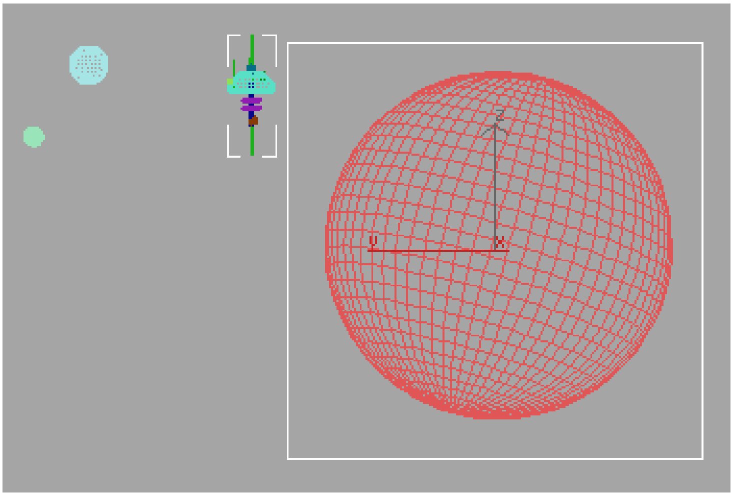
Take a few moments to familiarize yourself with the names of the objects in the scene. This will make it easier for you to select objects during this lesson.

Rotate Mars and its moons:

The first part of this lesson focuses on the three heavenly bodies you see in the scene. You will set up a dummy object to control the rotation of Mars and its moons, Deimos and Phobos.

1 In the Left viewport, create a Dummy object around Mars. Name the dummy object **MarsControl**.

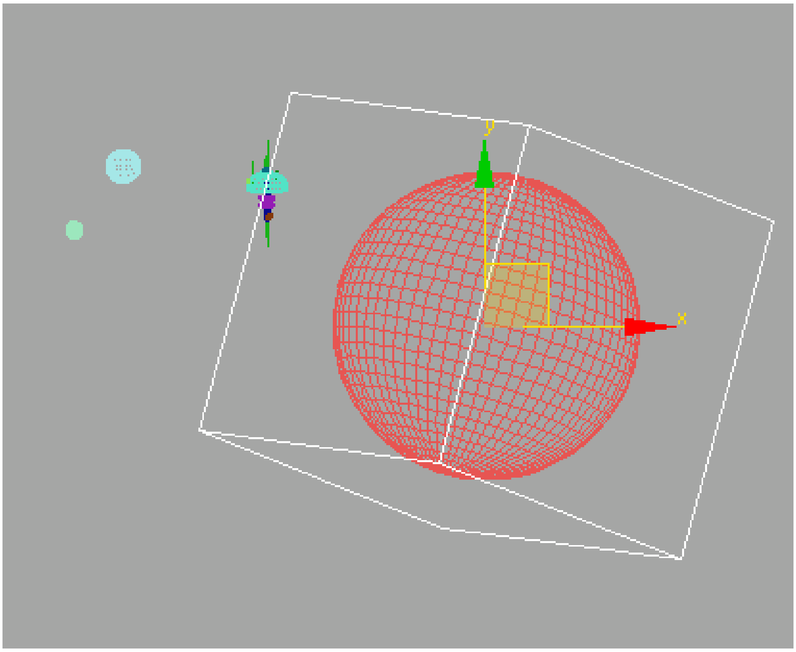
   Make the dummy a little larger than the planet so it's easier to pick.

2 With the dummy object still selected, choose Tools menu > Align > Align, and click *Mars*.

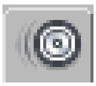
3 In the Align Selection (Mars) dialog, do the following:

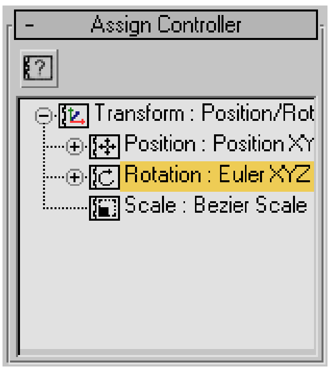
   - Turn on X, Y and Z Position in the Align Position (Screen) group.
   - Turn on X, Y and Z Axis in the Align Orientation (Local) group.
   - Click OK to accept the settings.



MarsControl is now aligned and oriented with the center of Mars.

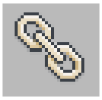
4 Select *MarsControl*.

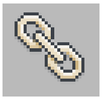
5  Go to the Motion panel, and expand the Assign Controller rollout. Select Rotation: Euler XYZ.



6  Click the Assign Controller button and choose TCB Rotation then click OK.

TCB Rotation will allow you to rotate objects on their Local axes as opposed to the World axes. This is beneficial when you have an object that is rotating on an axis that is tilted, such as the rotational axis of a planet.

7  Select *Mars*, then click Select And Link. Drag the rubber band to *MarsControl*. Release the mouse button when the cursor changes.

8  Link each of the moons, *Deimos* and *Phobos*, to *MarsControl*.

Mars and its two moons are now linked to *MarsControl*. Any movement or rotation you make to *MarsControl* will affect all the planetary bodies.

9  Click Select And Rotate, and select *MarsControl*.

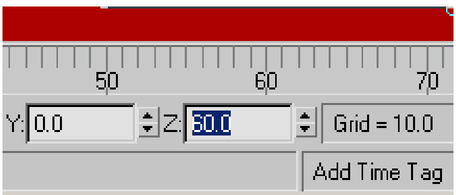
10 

Change the Reference Coordinate System from View to Local.

11 

Turn on the Auto Key button and move the time slider to frame 100.

12 In the Z axis field, below the time slider, enter **60**.



This rotates *MarsControl* by 60 degrees around its local Z axis. Because the planet and moons are linked to *MarsControl,* they also rotate.

13 Turn off the Auto Key and save your work as **MySpaceStation**.

14 Activate the SpaceCam viewport and play the animation.

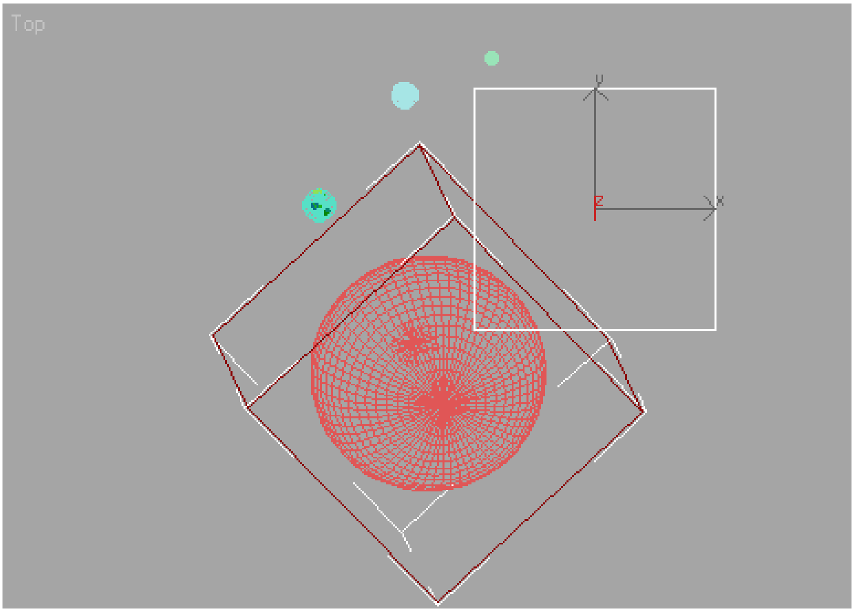
You will see Mars rotating on its axis, then at frame 60, Deimos swings into view and passes by and Phobos remains off-camera. If you like, you can zoom out to see both moons during playback.

15 Return the time slider to frame 0 before continuing.

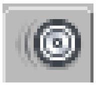
Set the space station into orbit:

Now that Mars is spinning on its own axis and Deimos and Phobos are orbiting Mars, you can set the space station into a geosynchronous orbit around Mars (an orbit that matches the planet rotation). You'll use the same technique for controlling the space station.

1 Add a new dummy object to the Top viewport, and name it **StationControl**.



It doesn't matter where you place the dummy object, because you'll align it to Mars in a few steps.

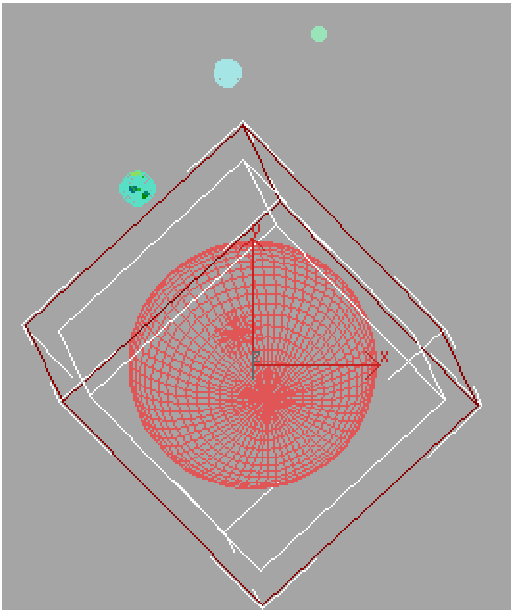
2  Open the Assign Controller rollout on the Motion panel, and select Rotation : Euler XYZ.

3  Click the Assign Controller button, and choose TCB Rotation. Click OK.

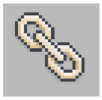
4 While *StationControl* is still selected, choose Tools menu > Align > Align, and click *Mars*.

5 In the Align Selection (Mars) dialog, do the following:

- Turn on X, Y and Z Position in the Align Position (Screen) group.
- Turn on X, Y and Z Axis in the Align Orientation (Local) group.
- Click OK to accept the settings.



These are the same settings you made when aligning *MarsControl* to Mars in the previous section.

6  In the Left viewport, link *SpaceStation* to *StationControl*.

7  Turn on Select And Rotate and select *StationControl*. Change the Reference Coordinate System from View to Local, if it's not already changed.

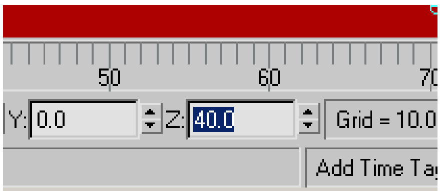
TIP You must always choose the transform (in this case, Select and Rotate) before choosing the Reference Coordinate System. Different transforms can have different Reference Coordinate Systems. If you choose the coordinate system first, it might change when you choose a different transform.

---

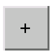
8 

Turn on the Auto Key button and move the time slider to frame 100.

9 In the Z axis field, below the time slider, enter **40**.



10 Turn off the Auto Key and save your work as **MySpaceStation01**.



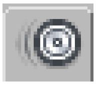
To create an incrementally saved file, use the Save As command from the Application menu, or click the plus-sign button.

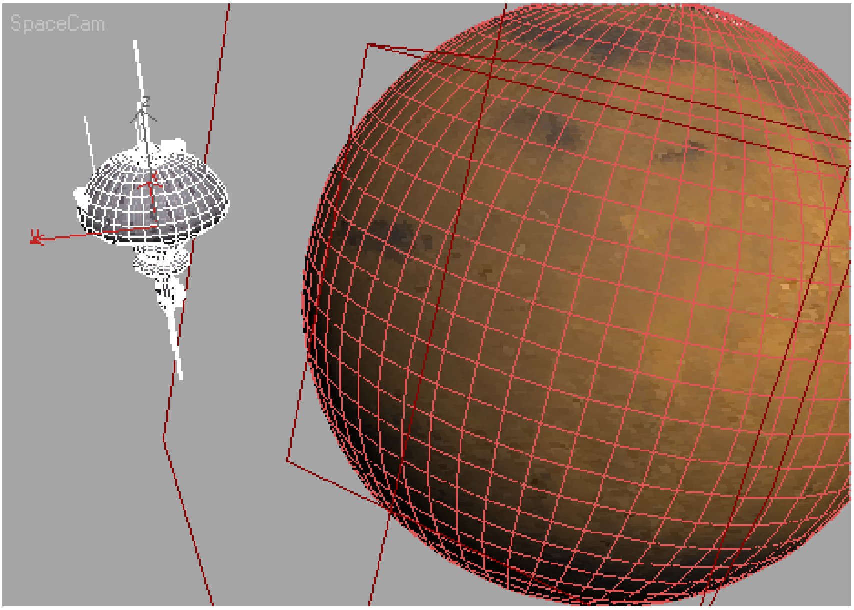
11 Play the animation.

Now the Space Station is orbiting around Mars but it's orbiting at a slower rate.

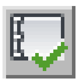
Add artificial gravity to the space station:

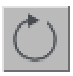
To generate some level of artificial gravity for its personnel, the space station must rotate around its own axis. This last section will solve that problem.

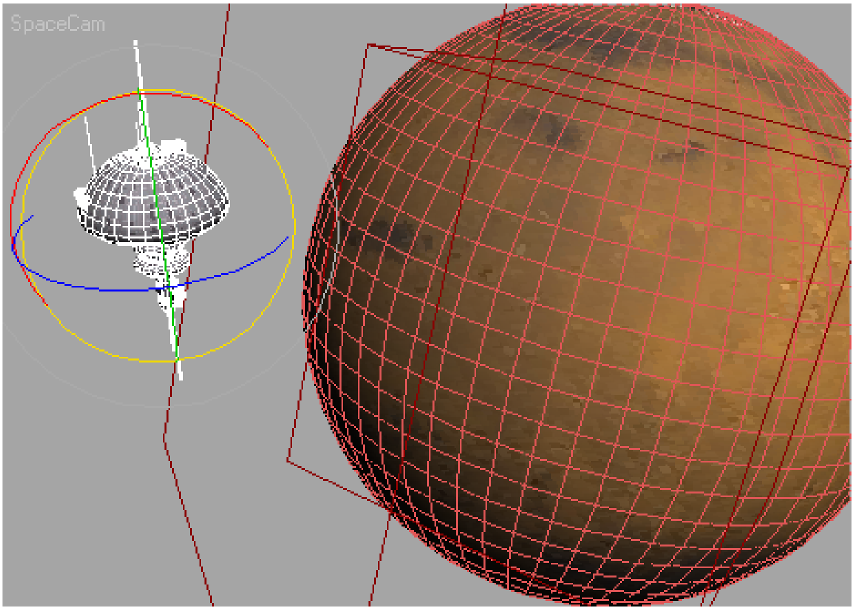
1  In the SpaceCam viewport, select *SpaceStation* and open the Motion panel.



2 Open the Assign Controller rollout and select Rotation : Euler XYZ.

3  Click the Assign Controller button and choose TCB Rotation, then click OK.

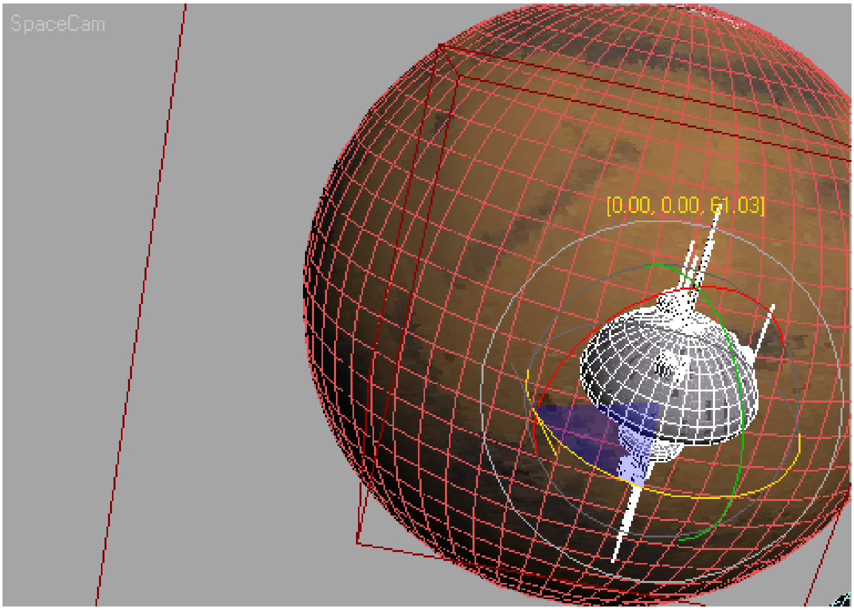
4  Turn on Select And Rotate if it's not already active. Set the Reference Coordinate System from View to Local.



5 

Turn on the Auto Key button and move the time slider to frame 100.

6 In the Z axis field, below the time slider, enter **90**.



7 Turn off Auto Key.

8 Play the animation.

Now the Space Station rotates about its own axis while it's in geosynchronous orbit around Mars. Maximize the SpaceCam viewport for a better view.

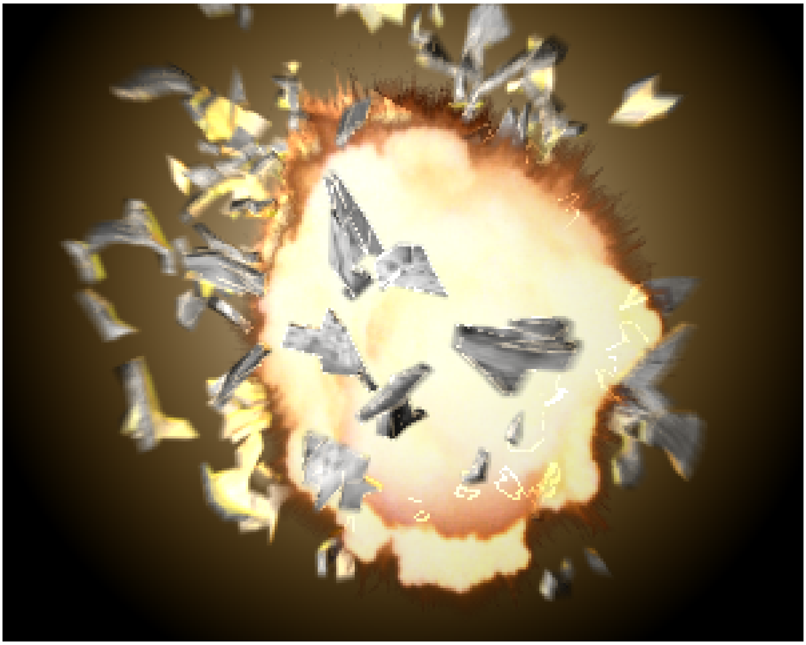
9 Save your work as **MySpaceStation02**.

Summary

This tutorial gave you a general introduction to using controllers and constraints to create and manage animation.

Creating Explosions

In this tutorial, you'll hit an asteroid with a glowing laser blast, blowing the asteroid to bits and creating a fiery explosion.



In this tutorial, you will learn how to:

- Use material effects channel IDs.
- Use animated opacity mapped plane objects to simulate an explosion.
- Use particle array fragmentation.
- Set visibility keys.
- Add motion blur.
- Use render effects glows and blurs.

Skill level: Intermediate

Time to complete: 2+ hours

Animating the Laser Blast

Animate the laser blast:

1  On the Quick Access toolbar, click the Open File button, navigate to the *\animation\exploding_asteroid* folder, and open the file *exploding_asteroid1.max*.

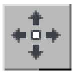
This scene already has an asteroid as well as a thin box object named *laserblast*.

2 Press H on the keyboard, highlight *laserblast* in the Select From Scene object list and click OK.

The laserblast object is selected, but not easy to see it in the viewport.

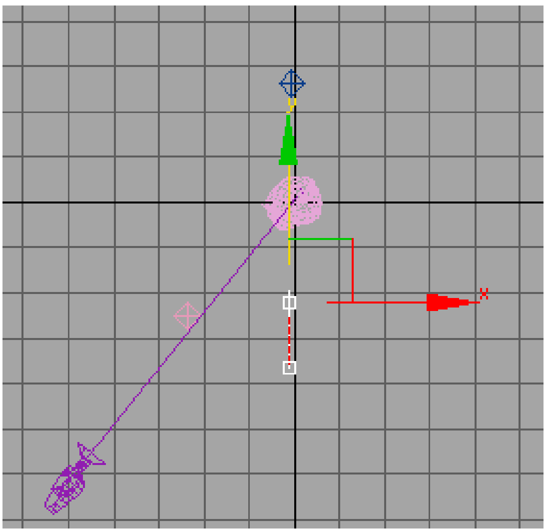
3 

Turn on Auto Key and move the time slider to frame 10.

4  In the Top viewport, move the *laserblast* along the Y axis until it comes in contact with the asteroid.

5 Turn off Auto Key.

A dotted line in the viewport shows the trajectory of the laserblast.



Move the time slider to frame 0, then play the animation in the camera viewport.

The *laserblast* moves to the asteroid and stops.

Creating a Material for the Laser Blast

Now you'll create a glowing material for your laser blast.

Create a material for the blast:

1.  Open the Material Editor and move it so you can see the camera viewport.
2. Move to frame 8 so you can see the *laserblast* clearly in the camera viewport.
3. In the Material Editor, select an unused sample sphere and name its material **laserblast material**.
4.  Click Assign Material To Selection.

   TIP If the *laserblast* object is no longer selected, you can drag the material from the sample slot into the viewport and onto *laserblast*.

5. In the Blinn Basic Parameters rollout > Self-Illumination group, turn off the Color check box. Set Self-Illumination to **100**.

   Self-illumination makes a material appear to glow. Non self-illuminated materials have a flat, unshaded look.

6. Click the Diffuse color swatch.

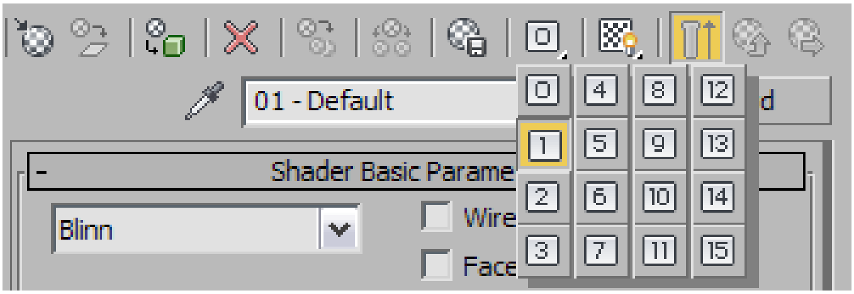
   The Color Selector dialog is displayed.

   Adjust the color to R:**255**, G:**114**, B:**0**, and click OK.

   The color of the material is a pumpkin orange.

Set up an effects channel:

1  On the Material Editor toolbar, click the Material ID Channel flyout, and choose channel 1.



Later in this tutorial, you'll use the channel number to create a glow.

2 Close the Material Editor.

Animating the Asteroid

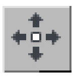
Now you'll animate the asteroid so it tumbles before it's hit by the laserblast.

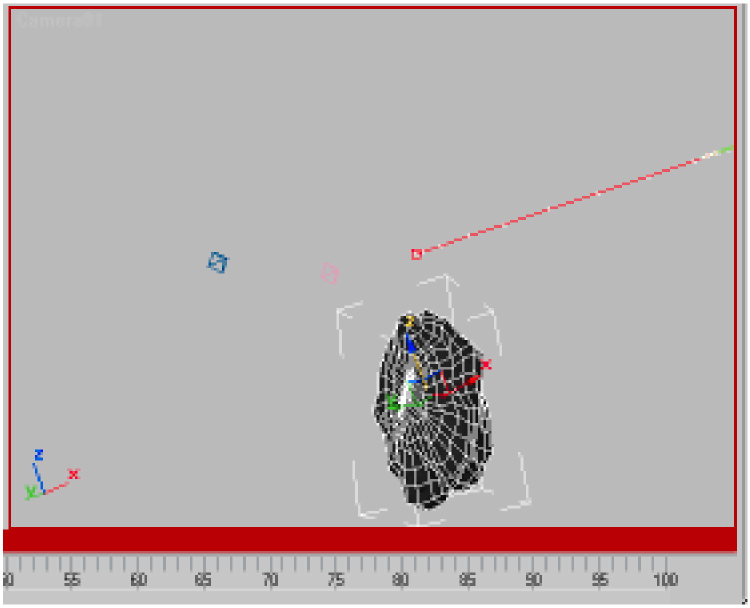
Animate the asteroid:

1 

Turn on Auto Key, if it is not already on.

2 Activate the Camera viewport and move the time slider to frame 0.

3  Select And Move the asteroid down in the Z axis approximately **–135** units, or until it is positioned just inside the camera's view.



Frame 0

4 Right-click and choose Rotate from the quad menu. Rotate the asteroid at frame 0 on both the Y and Z axes approximately **30** degrees.

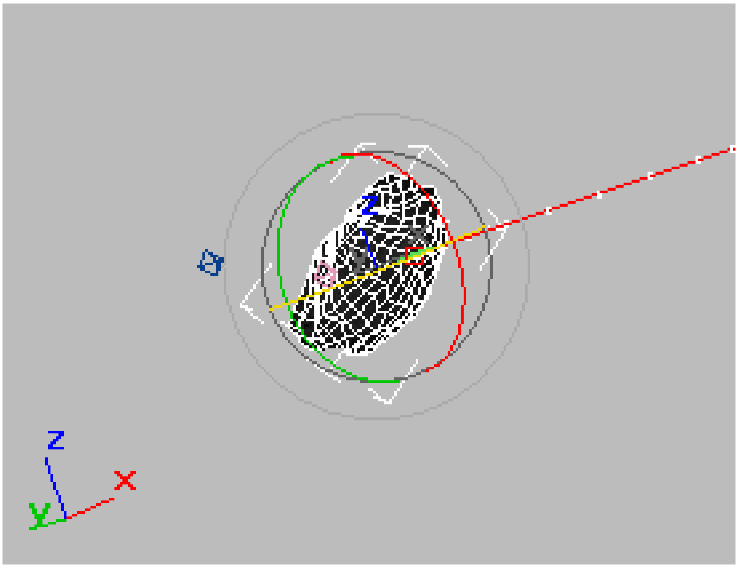
5 Go to frame 10. Reposition the asteroid so it is in contact with the *laserblast*. Rotate the asteroid on both the Y and Z axes approximately **–30** degrees.

6 

Turn off Auto Key.

7 Play the animation.

The asteroid now tumbles until it's hit by the laser blast.



Frame 10

8 Save your work as **myrotating_asteriod1.max**.

Creating a Plane for the Explosion

Now that the asteroid and *laserblast* are animated, it's time to create the explosion that will blow it up.

There are several different ways you can create an explosion in 3ds Max. One technique is to use an animated map. This allows you to apply a movie of a real explosion to a simple planar object.

Setup:

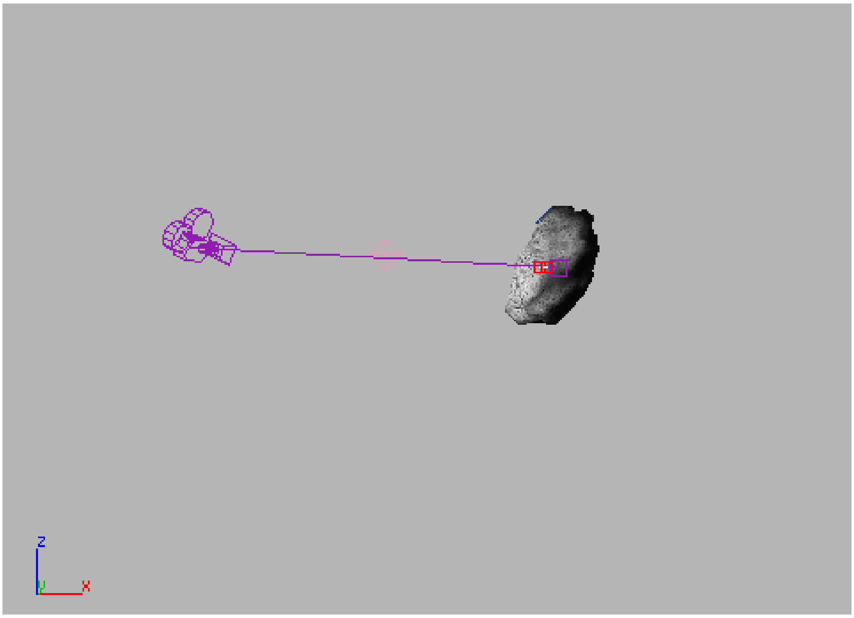
- Continue working with the previous file or open *rotating_asteroid.max*.

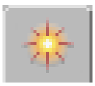
Create a plane:

1 Move the time slider to frame 10 if it's not already there.

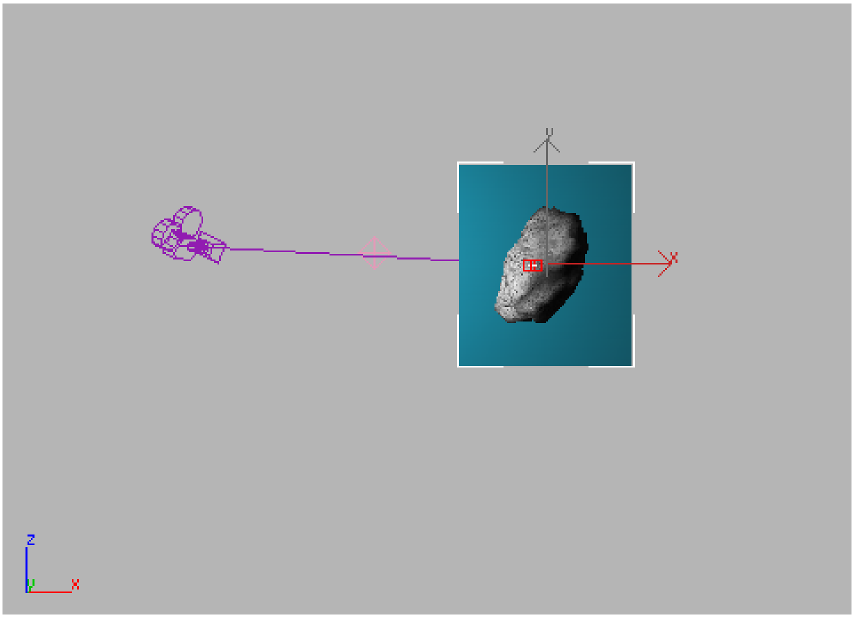
2 Activate the Front viewport, then press Alt+W on the keyboard to maximize it.

3   Use Zoom and Pan to navigate the viewport so the asteroid is in or near the center of the view.



4   Go to the Create panel > Geometry > Standard Primitives category. On the Object Type rollout, click Plane.

5 In the Front viewport, drag a plane over the asteroid.



TIP You can turn off snaps by pressing the S key. This is handy for toggling the snaps settings when you are in the process of creating or moving an object.

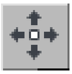
---

6 In the Name And Color rollout, change the name to **Explosion Plane01**.

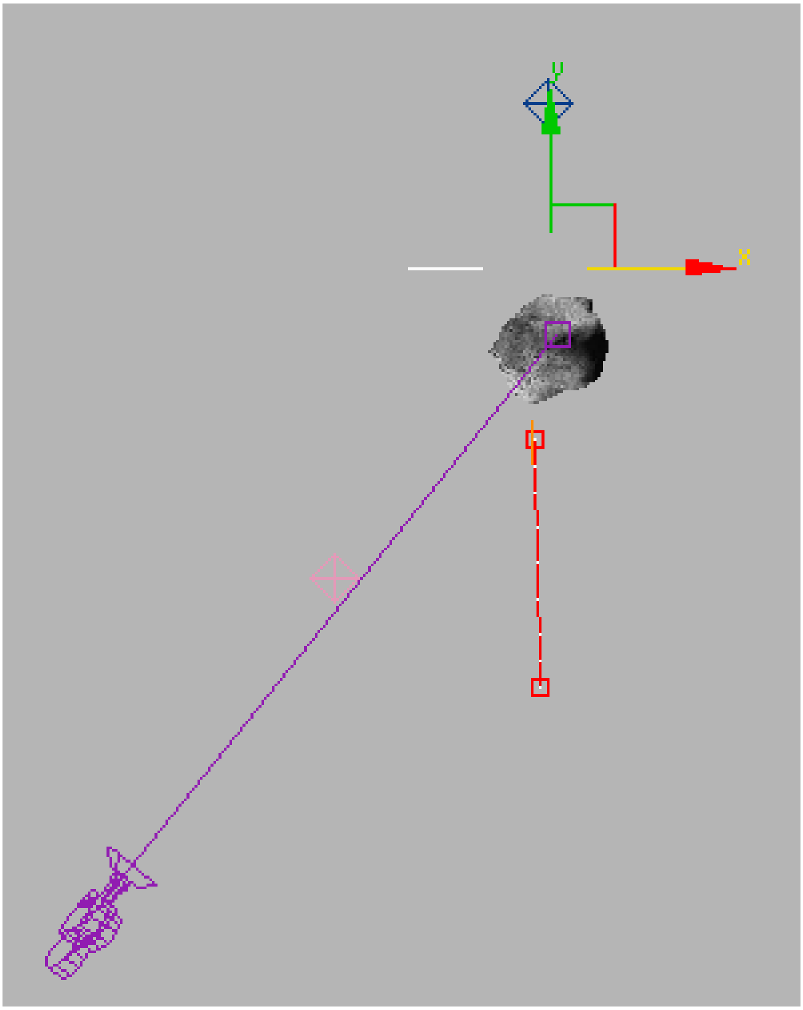
7 In the Parameters rollout, set the Length and Width of the plane to **300.0**. Set the Length and Width Segs (segments) to **1**.

  The size and complexity of the Plane update in the viewport.

8 Press Alt+W to return to four viewports.

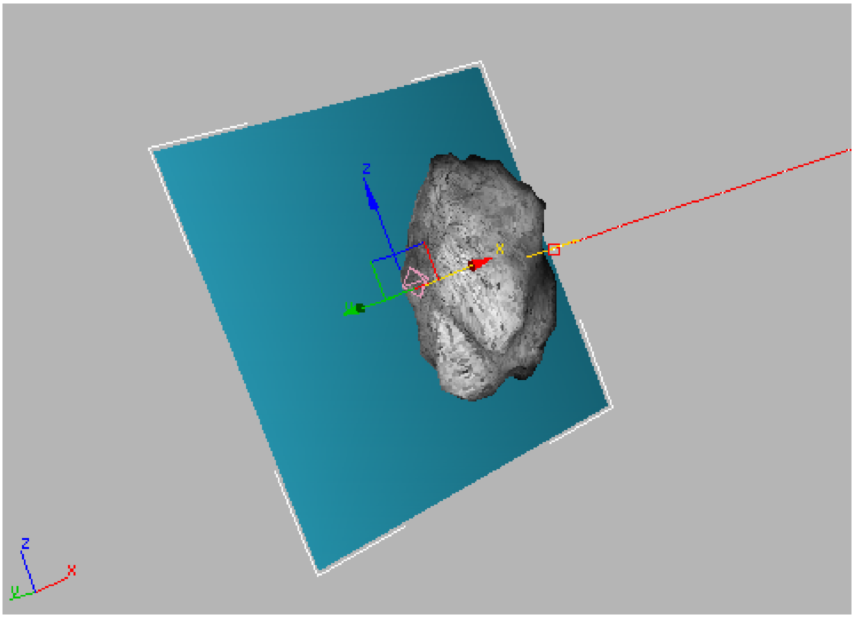
9  Turn on Select And Move. Press H and select the *ExplosionPlane01* object from the list, then activate the Top viewport.

10 Move the plane along its Y axis so it's behind the asteroid.



Moving the plane in the top viewport

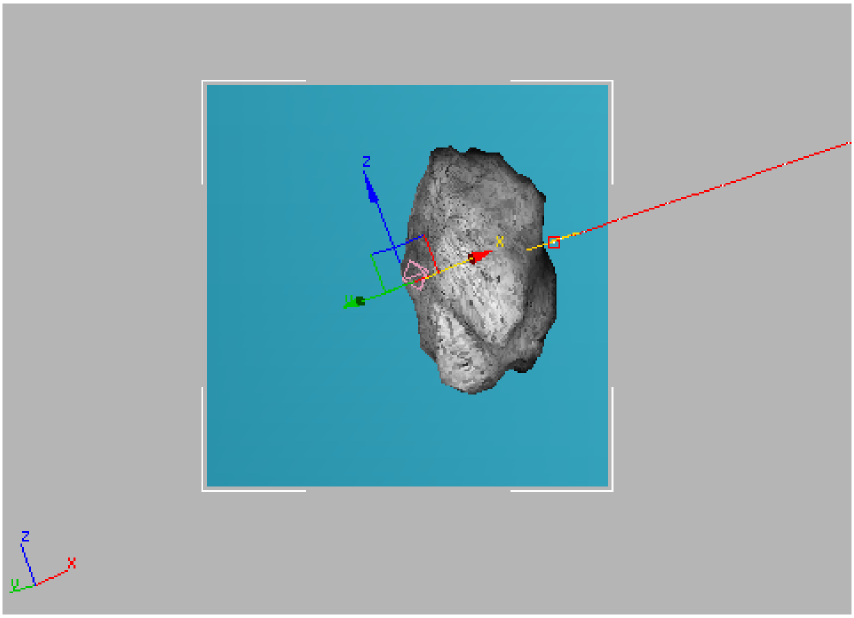
11 Right-click the Camera viewport to activate it without losing the selection of the plane.



Plane in camera viewport

12 From the Tools menu, choose Align > Align To View.

13 In the Align To View dialog, make sure Align Z is chosen, and click OK. The plane is now aligned to the Camera view.



Plane aligned to view

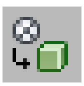
Creating a Material for the Explosion

Next, you'll create a material for the explosion.

Create a material for the explosion:

1  Open the Material Editor.

2 Select an unused sample sphere and name the material **Explosion**.

3  Click Assign Material To Selection.

TIP If the plane is no longer selected, you can drag the material from the sample slot to *Explosion Plane01*.

---

4 In the Blinn Basic Parameters rollout, click the blank button to the right of the Diffuse color swatch.

The Material/Map Browser dialog is displayed.

5 Click Bitmap in the dialog, and then click OK.

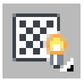
The Select Bitmap Image File dialog is displayed.

6 Use the Select Bitmap Image File dialog to find and choose *hercules.avi*.

The Preview window displays the first frame of the animation, which is solid black.

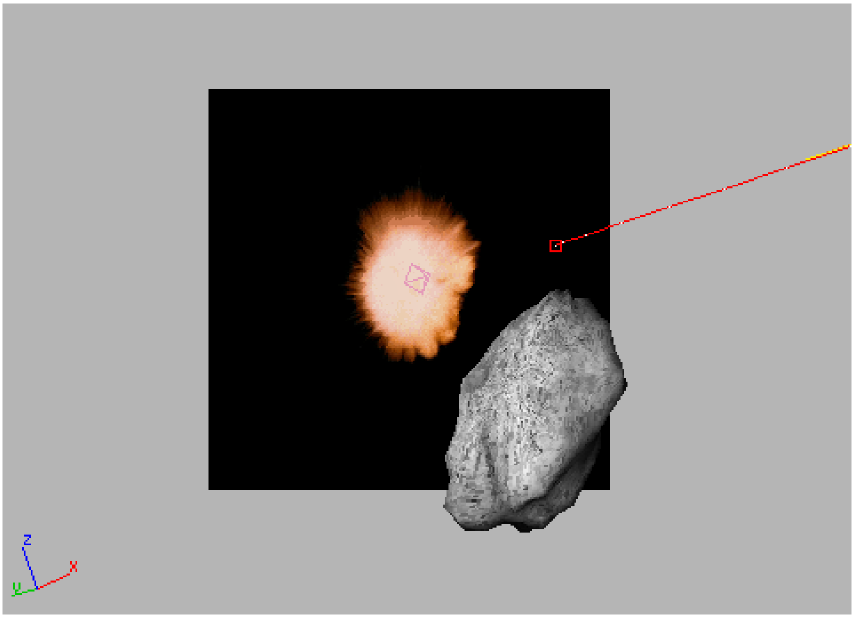
7 Click Open.

The animation is incorporated in the material as a Diffuse map.

8  On the Material Editor toolbar, click to turn on Show Standard Map In Viewport.

You can now see the frames of the animation on the plane object in the viewport.

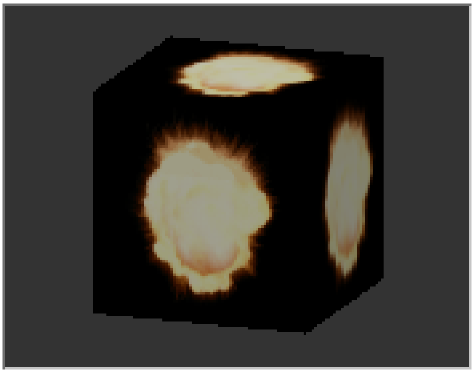
TIP To see the animation in the viewport, drag the time slider forward. (The first frame of the animation, which displays at frame 0, is black.)



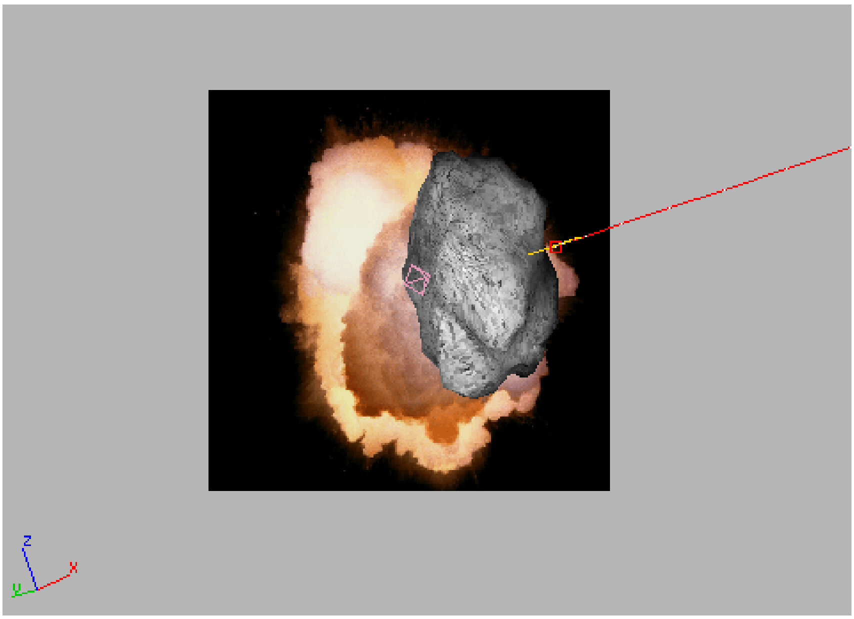
The material sample is currently a sphere. You can change that to a box to give you an undistorted view of the animated map.

9  On the Material Editor vertical toolbar (to the right of the sample slots), choose the Cube from the Sample Type flyout.

The Material sample is now a cube, not a sphere.



10 Move the time slider and watch the explosion play on the plane in the viewport.



Adding an Opacity Map

Now the explosion appears on the plane. The next step is to make the plane invisible so all you see in the scene is the explosion.

Add an opacity map:

1  In the Material Editor toolbar, click Go To Parent.

2 In the Material Editor, open the Maps rollout.

3 Click the Opacity map button (it is initially labeled “None”), and in the Material/Map Browser, choose Bitmap.

4 Use the Select Bitmap Image File dialog to choose *herculesm.avi*. This file is in the folder *sceneassets**images*\\. Click Open.

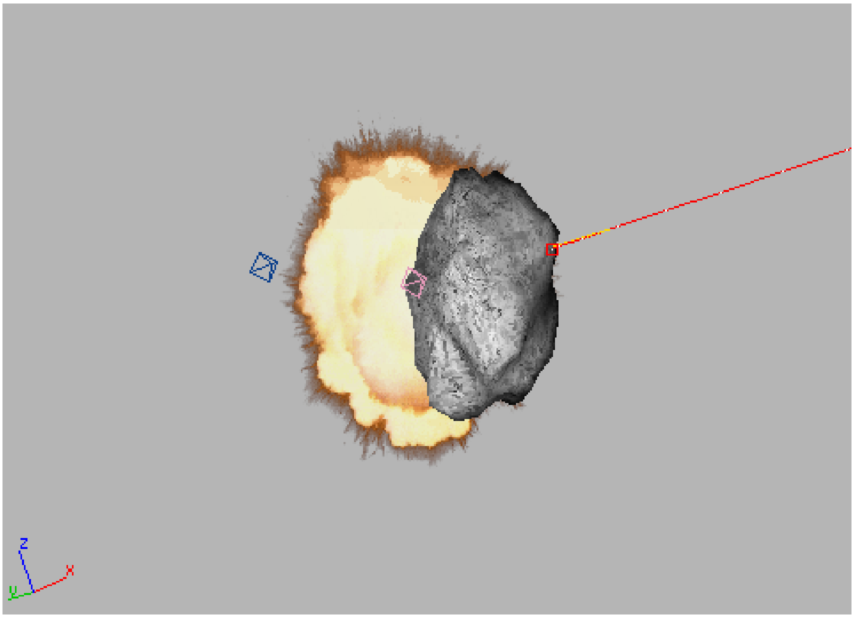
   The file *herculesm.avi* is a black-and-white mask of the animated explosion. Using the map’s alpha channel, the Opacity map makes the plane invisible and allows only the explosion to appear in the scene.

View the effect of the opacity map:

1  On the Material Editor toolbar, click Go To Parent. Close the Maps rollout.

2  On the Material Editor toolbar, click to turn on Show Standard Map In Viewport.

   The plane is now invisible in the viewport.



TIP If the transparency is not apparent in the viewport, click the plus (+) sign to the left of the viewport's label, choose Configure, and then in the Viewport Configuration dialog > Rendering Method panel, change Transparency to Best.

3  On the Material Editor vertical toolbar, click to turn on Background.

This turns on a checkered background that helps you visualize the opacity mapping.

Adjust the material settings:

- In the Blinn Basic Parameters rollout > Specular Highlights group, set both Specular Level and Glossiness to **0**.
  This removes any shininess from the plane.

  TIP To set a numeric field to **0**, right-click the spinner next to the field.

Adjust the plane properties:

When using this mapping technique, you don't want the invisible plane to cast or receive shadows, so you'll turn off those attributes .

1. Select the plane in a viewport, then right-click and choose Object Properties from the quad menu.
2. In the Object Properties dialog > Rendering Control group, turn off Receive Shadows and Cast Shadows. Click OK.

View the results:

- Play your animation.
  The plane doesn't move with the asteroid. That's okay, because you need the plane only at the point where the explosion occurs.

Synchronizing the Animated Maps

Here you'll set the timing so the animated explosion doesn't start until impact occurs at frame 10.

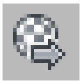
Set the timing:

1 Go to Material Editor > Maps rollout, and select the Diffuse map button, which is now identified as *hercules.avi*.

2 Open the Time rollout.

3 Change the Start Frame setting to **10**.

4 In the End Condition group, choose Hold.

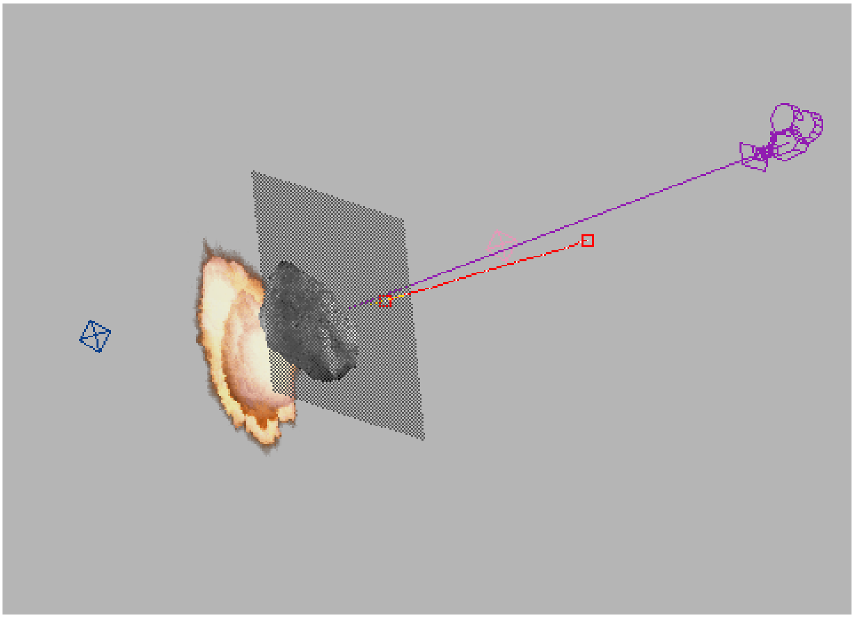
5  In the Material Editor toolbar, click Go Forward To Sibling to navigate to the next map in the material. In this case, it's the Opacity map.

6 Repeat steps 3 and 4 for the *herculesm.avi* opacity map.

7 Close the Material Editor.

8 Save your scene as **myexplosion1.max**.

Cloning the Explosion Plane



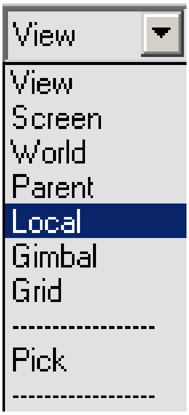
In this topic, you'll make a copy of the explosion plane, and set the copy in front of the asteroid. With two planes, the asteroid appears to be in the middle of the explosion, rather than in front of it.

Clone the explosion plane:

1 If it's not already selected, select *Explosion Plane01*. Press the Spacebar to lock your selection.

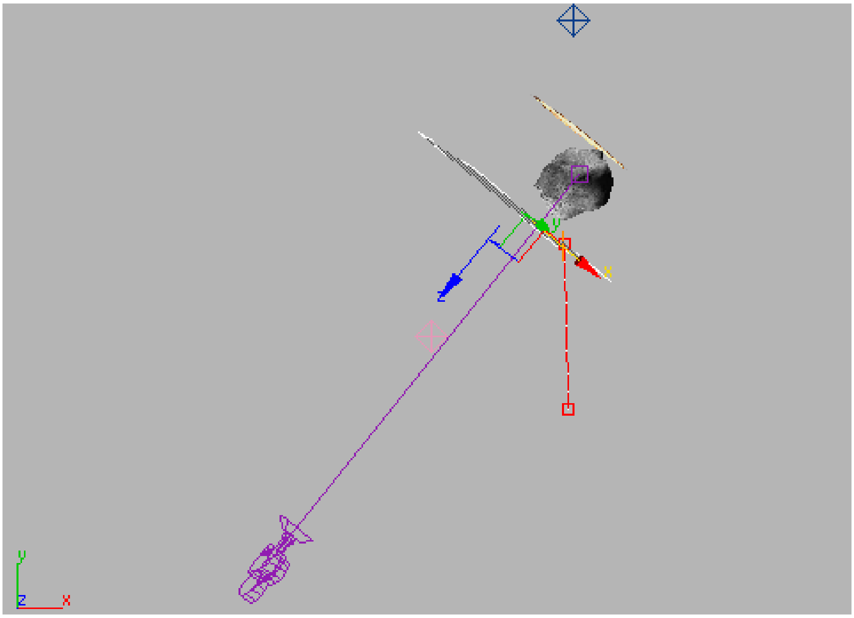
2 In the Top viewport, right-click the plane and choose Move from the quad menu.

3 On the toolbar, open the Reference Coordinate System list, and choose Local.



This will let you move the plane parallel to itself.

4 Hold down Shift and drag a copy of *Explosion Plane01* in the Z axis. Position the plane so it is in front of the asteroid.



Planes as seen in Top Viewport

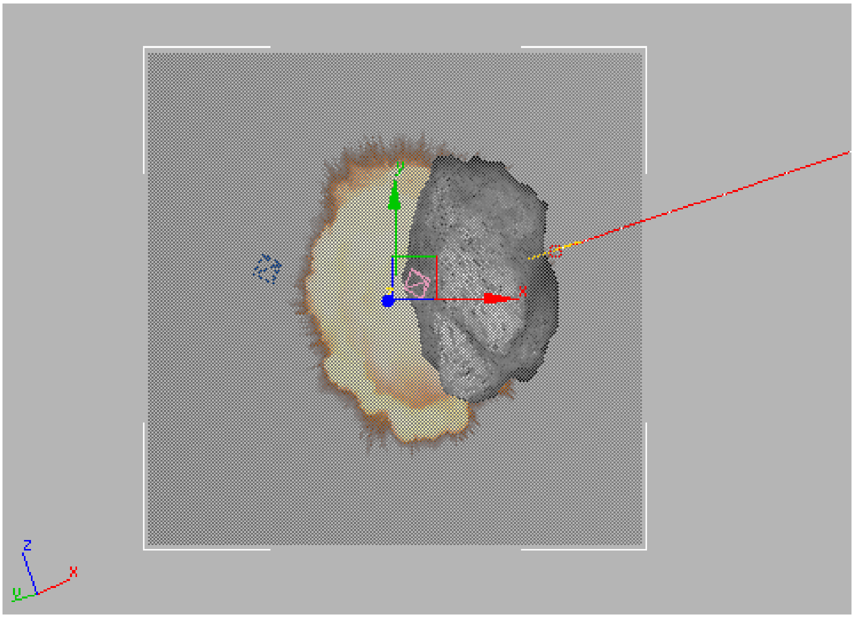
5 A Clone Options dialog is displayed. The program has automatically named the new plane *Explosion Plane02*. Leave the settings at their default, and click OK.

NOTE The copy of the plane is now selected, and the Selection Lock Toggle automatically turns off.

Set properties of the new plane, and make a named selection set:

1 Right-click the new plane and choose Object Properties. In the Display Properties group turn on See-Through. Click OK.

The cloned explosion plane now lets you see through it in the viewport. This doesn't affect the rendering.



See-through plane in camera viewport

2 Select the two plane objects. In the Named Selection Sets field on the toolbar, type the name **Explosions** and press Enter.

You can now reselect the two planes at any time by choosing this name from the Named Selection Sets drop-down list.

3 Right-click the selected planes and choose Hide Selection from the quad menu.

The two planes are hidden from view.

4 Save your scene as **myexplosion2.max**.

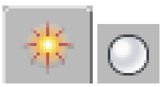
Breaking the Asteroid into Flying Pieces with Particle Array

At this point, the asteroid, *laserblast*, and explosion are all animated. Next, you'll add particles to simulate the asteroid breaking up as it explodes.

Setup:

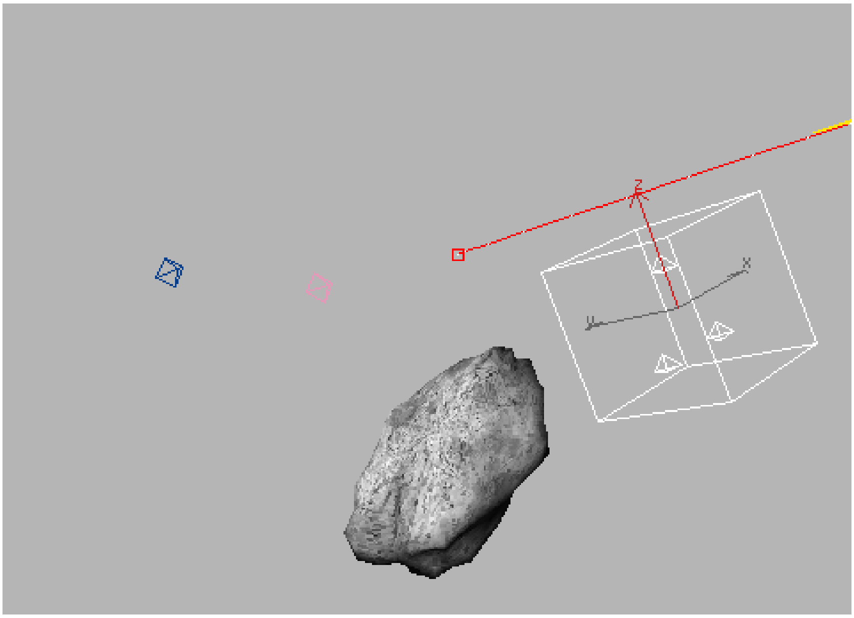
- Continue working with the previous file or open *exploding_asteroid_with_2planes.max*.

Create a particle system in the scene:

1  Go to Create panel > Geometry and choose Particle Systems from the drop-down list.

2 In the Object Type rollout, click PArray.

3 Click and drag to create a PArray particle gizmo in the Top viewport.



Perspective viewport view of the PArray gizmo

4 In the Basic Parameters rollout, click Pick Object, and then click the asteroid in a viewport.

   The name of the asteroid, in this case *Sphere01*, is displayed below the Pick Object button to show that the particle system has been linked to the asteroid.

Expand the command panel and adjust the PArray settings:

Next, you'll expand the command panel so you can see more of the particle system rollouts.

1 Move the cursor to the left edge of the command panel.

   The cursor changes to a double arrow.

2 Click and drag the edge of the panel to the left.

   A second column of the command panel appears.

3 Click the Particle Type rollout to open it.

   The Particle Type rollout is now displayed in the second column.

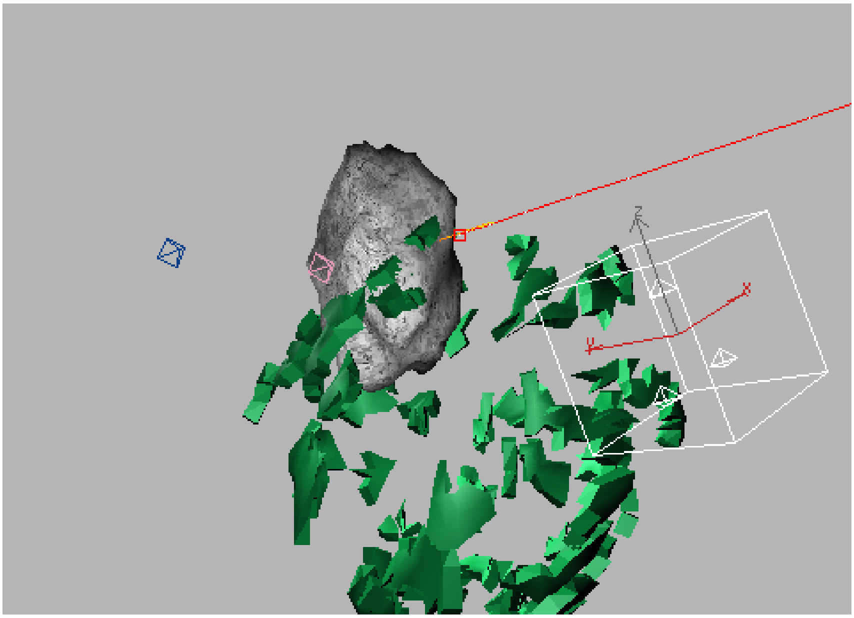
4 In the Particle Type rollout > Particle Types group, choose Object Fragments.

5 In the Object Fragment Controls group, set Thickness to **11.0**.

6 Choose Number Of Chunks and set Minimum to **75**.

7 In the Basic Parameters rollout > Viewport Display group, choose Mesh.

Move the time slider to see asteroid chunks appear in the viewport.

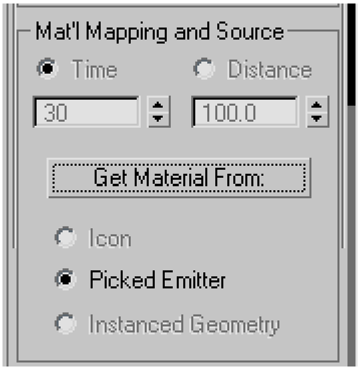


Adding Materials to the Particles

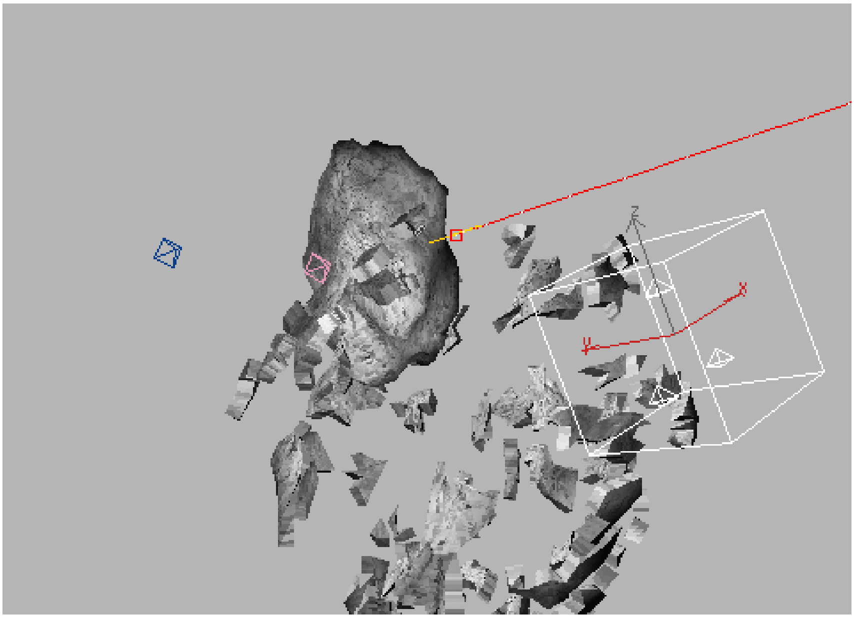
Add materials:

1 In the Particle Type rollout > Mat'l Mapping and Source group, choose Picked Emitter.

2 Click the Get Material From button.



The materials of the asteroid appear on the particles in the viewport.

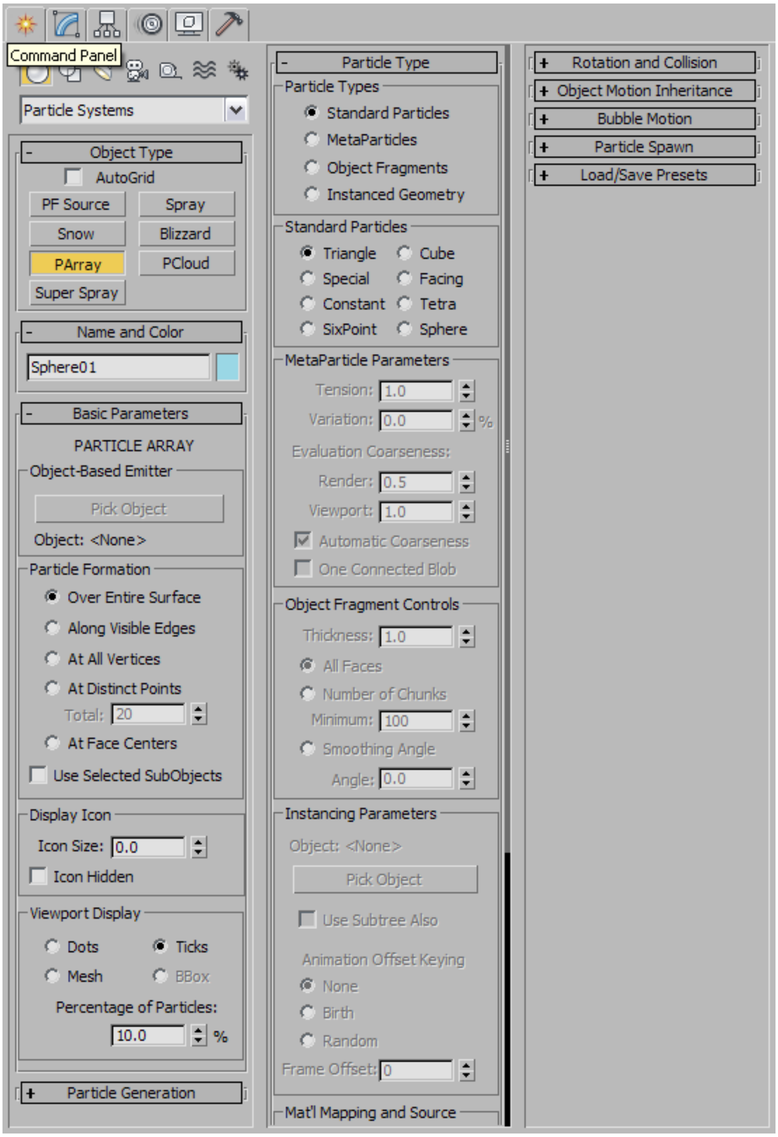


Expand the control panel again:

1 Move the cursor over the far-left edge of the command panel.
The cursor once again changes to a double arrow.

2 Click and drag the cursor to the left to open a third column.

Now the Basic Parameters rollout appears in the first column, the Particles Type rollout in the second, and most additional rollouts in the new third column.



TIP To close the additional columns, drag the left edge of the command panel to the right.

Controlling Particle Animation

Next, you'll add some control to the animated particles.

Control the animated particles:

1. Open the Rotation And Collision rollout. In the Spin Speed Controls group, set Spin Time to **50**.

   This will give some rotation to the chunks.

2. Open the Object Motion Inheritance rollout and set Influence to **0.0**.

   This will keep the asteroid movement and rotation from influencing the movement of the particles.

   By default, the particle animation will begin on frame 0. Because the asteroid doesn't blow up until frame 10, you'll have to adjust the timing of the particle animation.

3. Right-click an open rollout in the first column. Choose Particle Generation from the list.

   The Particle Generation rollout is now displayed.

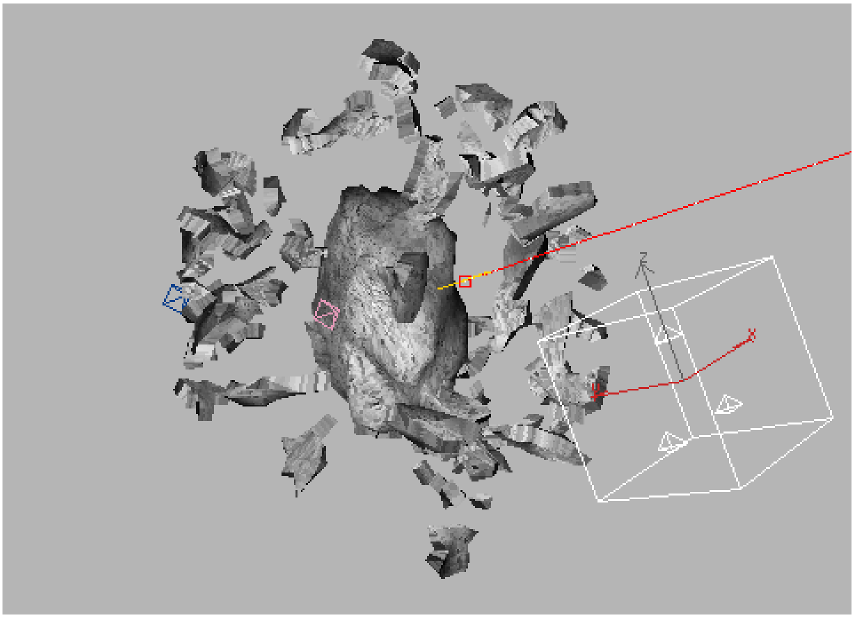
   TIP This right-click menu lets you navigate to a particular rollout quickly.

4. In the Particle Timing group, set Emit Start to **11** and Life to **90**.

   The particles don't appear until frame 11, and they persist until the last frame of the animation.

5. Move the time slider or play the animation in the viewport to see the effect.

   Now the asteroid starts to break into pieces after reaching the center of the viewport.



6 Drag the left edge of the command panel to the right to restore it to a single column.

7 Save your work as **myexplosion_w_particles.max**.

Setting Visibility Keys to Make the Asteroid Disappear

The asteroid particle system explodes into pieces; however, the original asteroid object is still visible in the scene. To make the explosion seem more realistic, the original object needs to disappear as it explodes.

Setup:

- Continue working with the previous file, or open *exploding_asteroid_with_particles.max*.

Create a visibility key:

1 

Go to frame 11 and turn on Auto Key.

2 In any viewport, select the asteroid then right-click and choose Object Properties from the quad menu.

3 In the Object Properties dialog > Rendering Control group, right-click the Visibility spinner arrows to set this value to **0.0**.

A red outline appears around the spinner arrows to indicate an animation key has been set.

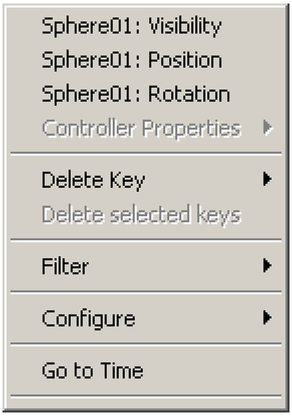
4 Click OK to close the Object Properties dialog.

5 

Turn off Auto Key.

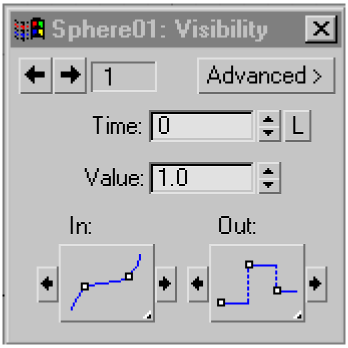
Adjust the new visibility keys:

1 In the track bar, under the time slider, right-click the key at frame 0. Choose Sphere01: Visibility.



2 If necessary, set the Value to **1.0**.

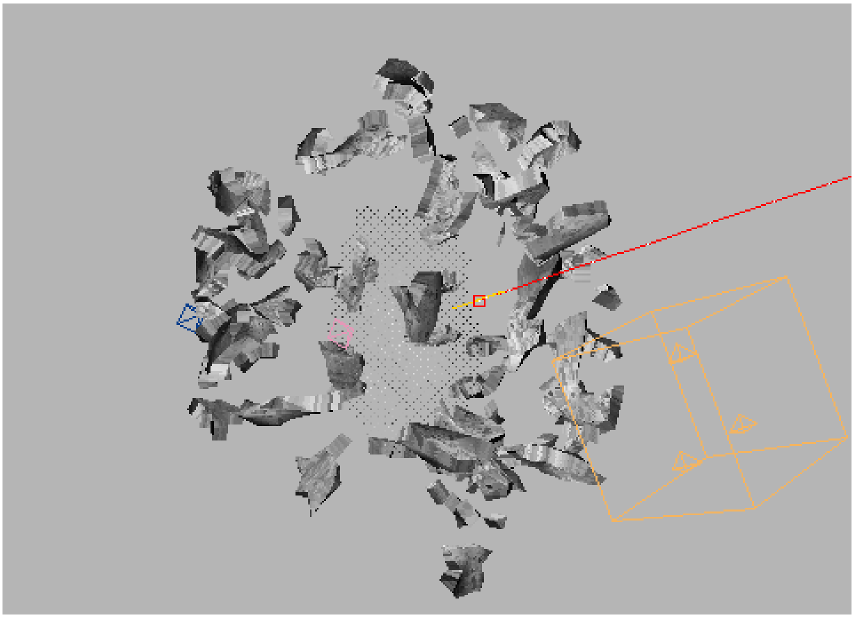
3 Click the Out interpolation icon and choose Stairstep from the flyout.



This will make the object visible until the next key.

4 Drag the time slider and observe the animation in the viewport.

The asteroid disappears and the fragments fly outward.



5 Close the Visibility dialog by clicking the X button in its title bar.

Render a frame to preview the explosion effect:

1 Right-click the camera viewport and choose Unhide All.

The planes are now visible again.

2 Go to frame 21.

3  On the toolbar, click Render Production.

The single image shows you what the effect will look like so far. Verify that the asteroid is invisible, and that only the particle fragments and the explosion planes are rendering.

Adding Motion Blur

To help create the illusion of fast movement in your explosion, you'll add motion blur to your animation.

Setup:

- Close the rendered frame window, if it is still open.

Add motion blur:

1 At frame 0 select the asteroid, then right-click it and choose Object Properties on the quad menu.

2 In the Motion Blur group, make sure that Enabled is on, and choose Object. Click OK.

   Object motion blur will blur the asteroid, but not the entire scene.

3 Move the time slider to frame **7**.

4  Activate the Camera viewport and click Render Setup on the main toolbar.

   The Render Setup dialog is displayed.

5 Click the Renderer tab. On the Default Scanline Renderer rollout > Object Motion Blur group, make sure that Apply is on, and that Duration (Frames) is set to **0.5**.

   This will create a slight motion blur.

6 At the bottom of the dialog, click Render to test the result.

Motion blur on asteroid

7 Save your scene as **myexplosion3.max**.

Putting a Light Inside the Asteroid

Creating fiery explosions with mapping techniques is effective, but it doesn’t actually illuminate your scene. Adding a light inside the exploding asteroid illuminates its fragments and creates a more dramatic effect.

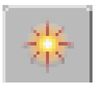
Setup:

- Continue working with the previous file or open *exploding_asteroid_with_motionblur.max*.

Put a light inside the asteroid:

1 Move the time slider to frame 10.

This puts the asteroid in the center of the frame.

2   Go to Create panel > Lights > Object Type rollout, choose Standard from the drop-down list, and then click Omni.

3 Activate the Top viewport, then click the center of the asteroid to create an Omni light.

Because the asteroid is positioned at the center of the scene, the light appears inside the asteroid.

4 In the Name and Color rollout, change the name of the light you just created to **inside asteroid light**.

5 Go to the Modify panel and click the white color swatch in the Intensity/Color/Attenuation rollout.

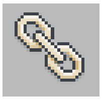
The Color Selector is displayed.

6 Change the color to orange (R: **255**, G: **111**, B:**56**). Then close the Color Selector.

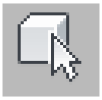
7 Set the Multiplier to **4.0**.

NOTE You don't need to animate the light. It will illuminate only the particles, and they appear when the explosion begins.

---

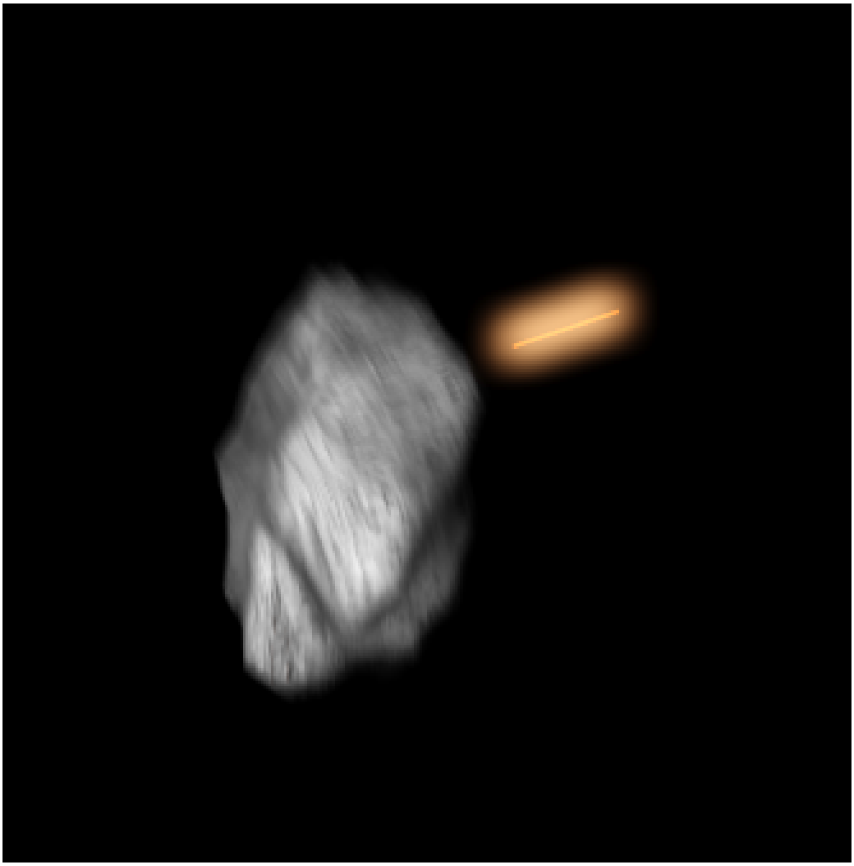
8.  On the toolbar, turn on Select And Link. Then press the H key to select the parent object. Choose *Sphere01* from the list, and then click Link.

   Now the light will move along with the animated asteroid.

9.  On the toolbar, click the Select button to turn off Select And Link.

10. Save your work as **myexplosion_w_light.max**.

Adding Render Effects to the Laser Blast



To give your scene some finishing touches, you'll add render effects to make the laser blast and the explosion glow.

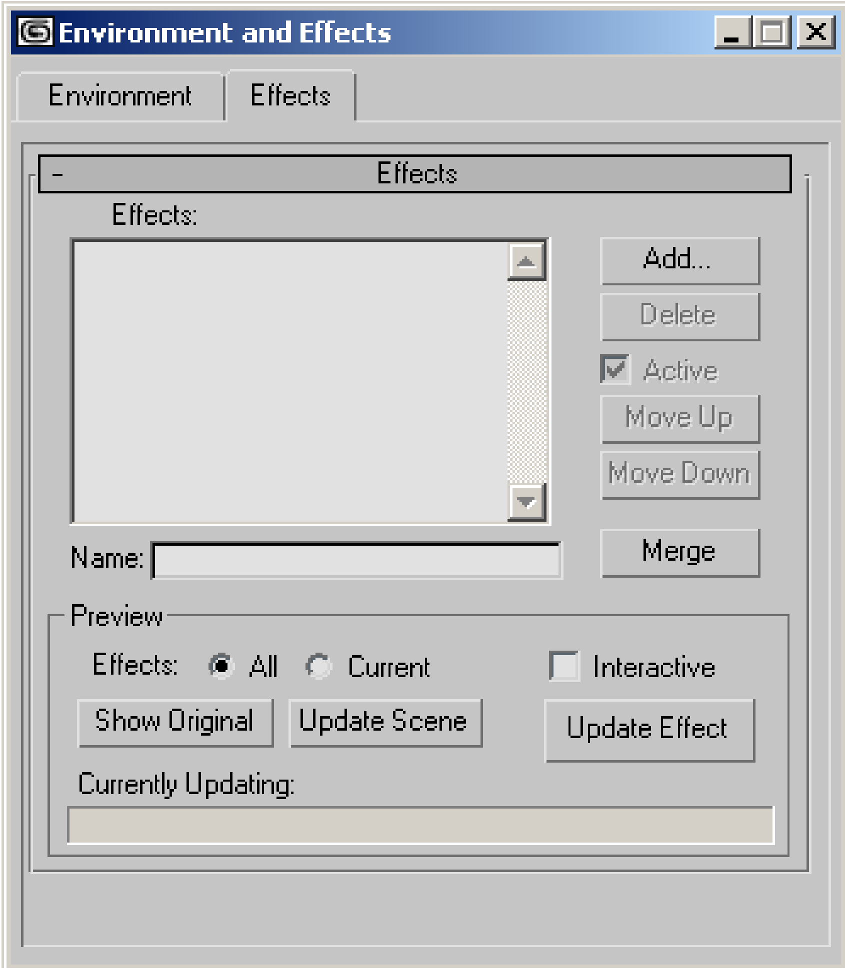
Setup:

- Continue working with the previous file or open *exploding_asteroid_with_lights.max*.

Add a render effect:

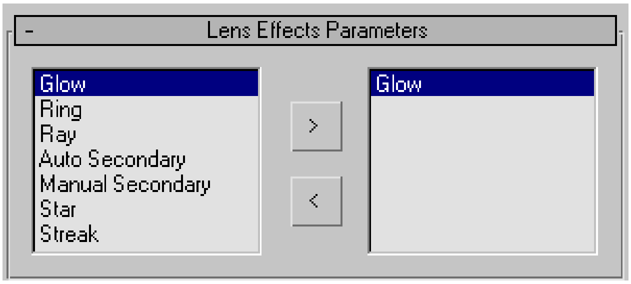
1 Choose Rendering menu > Effects.

The Environment And Effects dialog is displayed, with the Effects tab active.



2 Click Add. In the Add Effect dialog, choose Lens Effects from the list, and click OK.

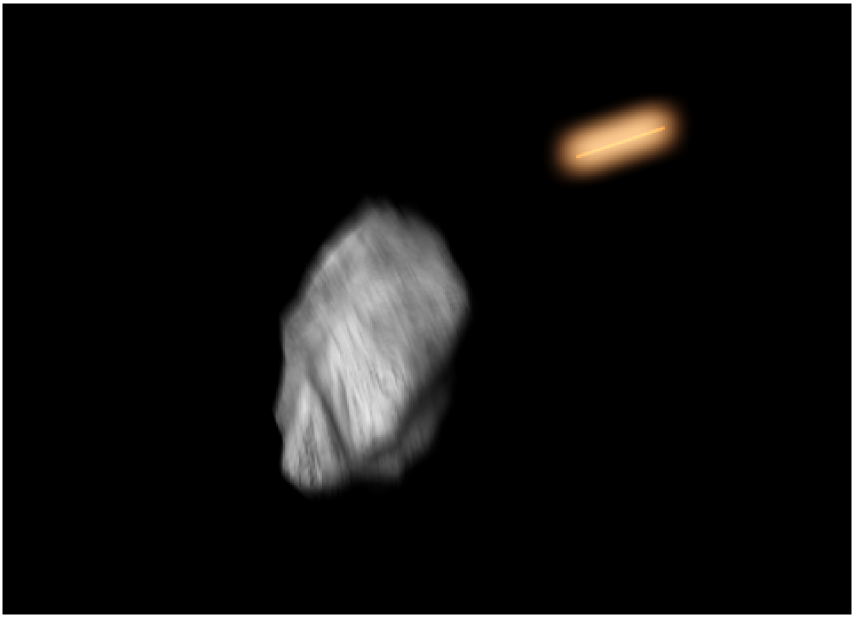
3 In the Lens Effects Parameters rollout, highlight Glow and then click the right-pointing arrow to move it into the list on the right.



Adjust the glow settings:

1. Scroll down to the Glow Element rollout, and in the Name window, rename the glow effect **laser glow**.
2. Change Size to **0.25** and Intensity to **200**.
3. In the Radial Color group, click the white color swatch and change the color to orange (R: **235**, G: **120**, B: **60**).
4. Click the Options tab.
5. In the Apply Element To group, make sure Lights and Image Centers are off.
6. In the Image Sources group, turn on Material ID. It should be set to **1** by default.
7. Move the time slider to frame 5.
8. Scroll up to the Effects rollout > Preview group. Make sure the *Camera01* viewport is active, then click the Update Scene button.

   The frame renders, and then the glow is applied to the laser blast.



9 Save your scene as **myexplosion4.max**.

Adding a Second Glow to the Explosion

Next, you'll add a glowing effect to the light inside the asteroid to give the blast added realism.

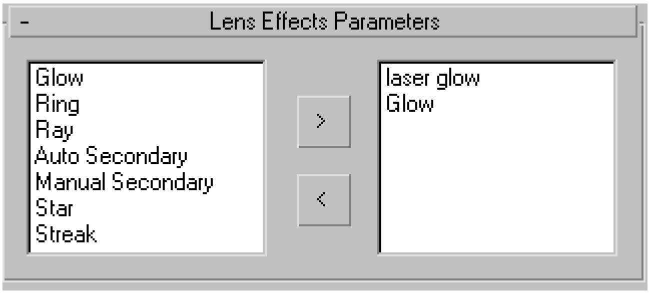
Setup:

- Continue working from the previous lesson, or open *exploding_asteroid_with_laserglow.max*.

Add a glow effect:

1 Choose Rendering > Effects, and highlight Lens Effects in the Effects list.

2 On the Lens Effects Parameters rollout, choose Glow once again in the list window on the left and click the right-pointing arrow.

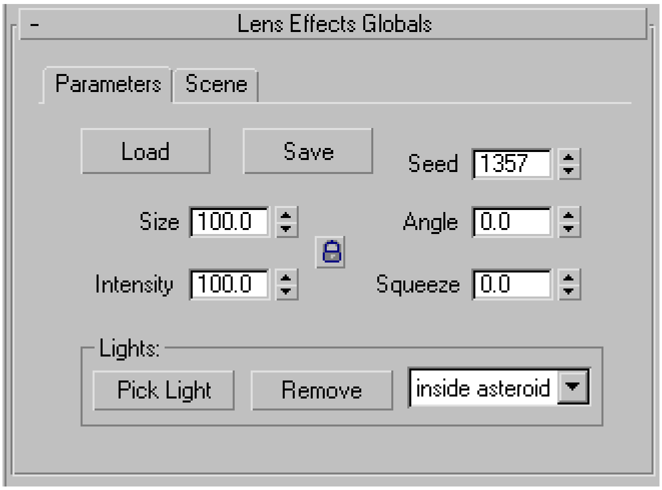


3 Scroll down to the Glow Element rollout and name the glow effect **blast**.

Adjust the glow settings:

1 In the Lens Effects Globals rollout, turn on Pick Light.

2 Press H and in the Pick Object dialog, select *inside asteroid light* and click Pick.



The name of the light is now displayed in the text field.

3 On the Glow Element rollout, open the Options panel.

4 In the Apply Element To group, turn on Lights. Turn off Image and Image Centers, if activated.

5 Go to the Parameters panel, and set Occlusion to **0.0**.

This allows you to see the glow through the explosion planes in the scene.

6 Turn off Glow Behind.

7 In the Radial Color group, change the swatch on the right to an orange color (R:**242**, G:**150**, B:**0**).

8 In the Glow Element > Parameters panel, set the Size to **30.0**.

9 Set the Intensity to **50**. Keep the Environment and Effects dialog open.

Preview the effect:

1 Go to frame 25 and activate the *Camera01* viewport.

2 Scroll up to the Effects rollout. Click Update Scene to see the results.

3 Close the rendered frame window and the Environment and Effects dialog.

Animating the Explosion's Glow

The glow will be more realistic if it changes over time.

Animate the explosion's glow:

1  On the toolbar, click Curve Editor (Open).

The Track View - Curve Editor is displayed.

2 Go to the Modes menu and choose Dope Sheet.

3 In the Controller window on the left, pan down until you can see the label Render Effects. This is near the top of the scene hierarchy.

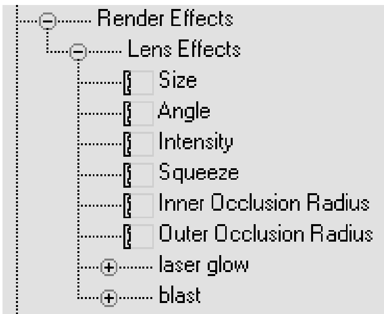
4 Click the plus icon to the left of Render Effects to expand the render effects tracks.

TIP Clicking a plus icon in this window expands tracks. When the tracks have been expanded, the icon changes to a minus icon. If you want the tracks collapsed, click the minus icon. The tracks will no longer be visible.

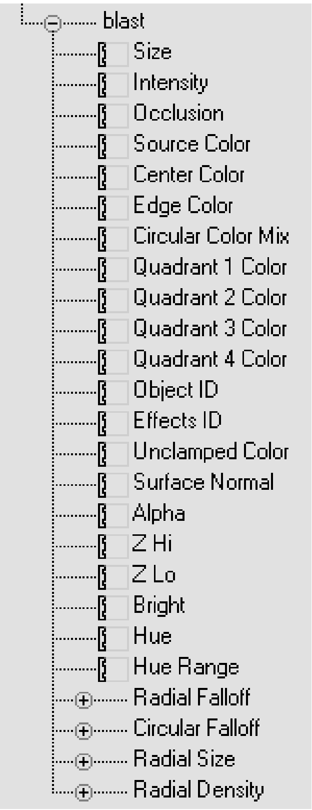
---

5 Click to expand the Lens Effects tracks.

With the Lens Effects tracks open, you can see both glow effect tracks.

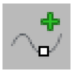


6 Scroll down and click to open the *blast* tracks.



All the glow attributes in the Rendering Effects dialog are listed here. You can set keys for any attribute to animate its effect.

7 Click the Intensity track to select it, then right-click and select Assign Controller. Select Bezier Float, and then click OK.

8  With the Intensity track selected, turn on Add Keys on the Track View toolbar.

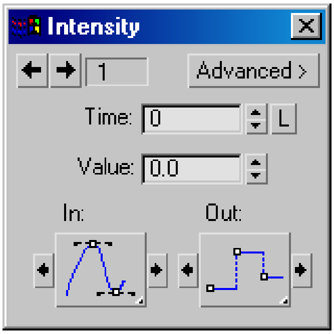
9 In the Key window on the right, click the Intensity track to add a key at frame 0 and another key at frame 9.

TIP When you move the time slider, the vertical blue lines in the Dope Sheet indicate the current frame. This is a useful way to find the frame you need.

---

10 Right-click the first key and set its value to **0**. Move to the second key and set it to **0** as well.

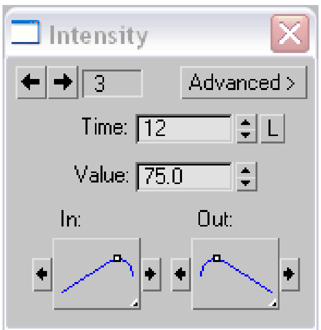
11 Set the Out interpolation value of the first key to Stairstep. This will lock the glow between the first two keys.



12 Add another key at frame 12. Right-click the key and set its value to **75.0**.

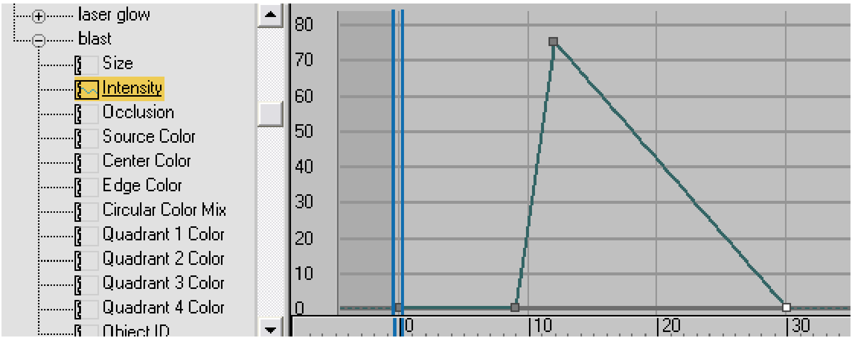
13 Add another key at frame 30. Right-click the key and set its value to **0**.

14 At frame 9, set the interpolation to Linear for the Out value keys. At frame 12, set the interpolation to Linear for In and Out values, and at frame 30, set Linear for the In value.



15 Close the Intensity dialog by clicking the X button in its title bar.

16 You can see the shape of the intensity curve you just created by choosing Modes > Curve Editor. (You might have to expand tracks and scroll to find the blast > Intensity track once again.)



Animated intensity displayed in the Curve Editor

When the explosion begins, the Intensity rapidly rises to 75.0, then more slowly fades until at frame 30 it is zero once again.

Adding Streaks with Radial Blur

You can get a nice streaking effect using the Blur render effect. Adding a blur is just like adding a lens effect.

Add a blur effect:

1 In the *Camera01* viewport, select the PArray gizmo, right-click it, and choose Object Properties from the quad menu.

2 In the G-Buffer group, use the spinner to set the Object ID to **2**. Click OK.

3 Choose Rendering menu > Effects.

4 Click the Add button, choose Blur, and click OK.

Set up the blur effect:

1 In the Blur Parameters rollout > Blur Type panel, choose Radial.

2 On the Pixel Selections panel, turn off Whole Image, and turn on Object ID.

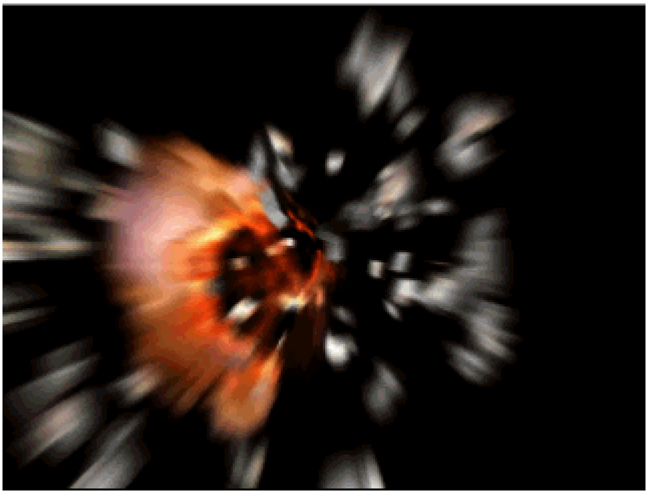
3 In the Object ID controls, use the spinner to change the ID to **2**, then click the Add button.

   The number 2 appears in the Object ID list.

4 Activate the *Camera01* viewport and drag the time slider to frame 33.

5 In the Effects rollout, click Update Scene to preview the result.

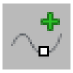
   The radial blur is added to the image.



Animate the blur effect:

Finally, you will keyframe the values and interpolation of the Blur Radial Pixel radius, so the particles fade out as they slow down toward the end of the animation.

1 Open the Dope Sheet Editor again and in the Controller window, navigate to the Render Effects > Blur entry. Expand its tracks, and click to highlight Blur Radial Pixel Radius.

2  Turn on Add Keys on the Track View toolbar. In the Key window, click to add keys at frames 0 and 11.

3 Right-click a Blur Radial Pixel Radius key. Give both these keys a value of **0.0**, and use stairstep interpolation between them.

4 Click to create a key at frame 18. Give it a value of **7.0**,. Leave the interpolation set to the default of Bezier.

5 Set another key at frame 100 with a value of **1.0.**

6 Save your scene as **myexplosion5.max**. You can open *exploding_asteroid_final.max* for comparison

Render the animation:

1 On the Rendering menu, choose Render Setup. Make sure that the Common tab is displayed; click it if it isn't.

2 On the Common Parameters rollout, in the Time Output group, turn on Active Time Segment: 0 to 100.

3 In the Render Output group, click the Files button. Name the output file of the finished explosion **myexplosion2.avi**, and press Enter.

4 In the AVI File Compression Setup dialog, click OK.

5 Click Render at the bottom of the Render Setup dialog.

Or you can just play the provided movie *exploding_asteroid_final.avi*.

TIP When doing high-quality work, render to a still image sequence of TGA files. Then you can use the Ram Player or other methods to turn these files into a movie sequence such as AVI or MOV.

Summary

In this tutorial, you created an exploding asteroid and learned how to combine the use of many tools to create this effect. You learned how to use material effects IDs with glows, and created explosion planes with animated opacity mapping. You've created exploding fragments using PArray objects, and learned how to set visibility keys. Finally you've used Render effects and motion blur to complete the shot.

Working with the Walkthrough Assistant

Instead of manually manipulating the camera, you can make animating much easier by using the Walkthrough Assistant. The Walkthrough Assistant allows you to “fly” your camera to explore your virtual designs. Although the tool's name implies interior usage, the Walkthrough Assistant is actually suitable for both interior and exterior scenes.

In essence, it simplifies the tasks of creating a camera, constraining that camera to a path of your choice, and then makes it easy to edit and animate various aspects of the camera such as tilt, pan and lens values. All manipulation is centralized in the same dialog so you do not have to browse through various panels to make changes to camera parameters.

In this tutorial, you will learn how to:

- Create a target camera
- Adjust camera head tilt and angle
- Use Set Key to animate a camera
- Render an animation to a sequence of still image files
- Create a movie file of your animation

Skill level: Beginner

Time to complete: 1+ hours (includes rendering time to create an animation.)

Camera Setup with Walkthrough Assistant

In this lesson, you'll use the Walkthrough Assistant to animate a camera, simulating a walk or run along the Great Wall of China. The scene contains a camera path ready for constraining the camera, but you will first need to create a sufficient number of frames to hold your animation.

Set up for this lesson:

-  On the Quick Access toolbar, click the Open File button and from the *\animation\walkthrough_asst* folder, open *great_wall_start.max*.

TIP If the Units Mismatch dialog displays, choose Adopt The File's Unit Scale and then click OK.

Calculate the number of frames:

1 In the right viewport, select the blue spline that represents the camera path. Alternatively, press H to open the Select From Scene dialog, and double-click Camera Path.

2  On the Utility panel, click the Measure tool.

This tool reports the length of the camera path is roughly 900 feet.

NOTE For a comfortable walking pace, which is useful in architectural walkthroughs, you'll need about a second for every 3 feet of distance. For a jog or a fast run, you can go as far as 9 feet for a second. In NTSC format, that translates into 30 frames for every 9 feet of distance traveled, or 3000 frames for 900 feet.

3  Click the Time Configuration button next to the Current Frame Field.

4 In the Animation group in the Time Configuration dialog, change Start Time to **1**. Change End Time to **3000** to increase the number for frames in the animation, and then click OK.

This will provide sufficient frames for your walkthrough animation.

The time slider frame indicator now displays 3000 frames.

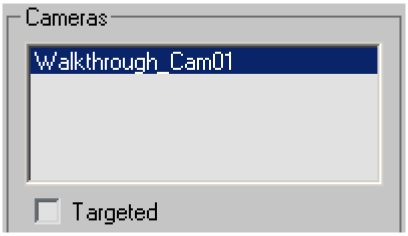
Creating a basic setup using Walkthrough Assistant:

1 From the Animation menu, choose Walkthrough Assistant.

A modeless dialog displays.

2 In the Camera Creation group in the Main Controls rollout, ensure the Free Camera option is chosen, and then click Create New Camera.

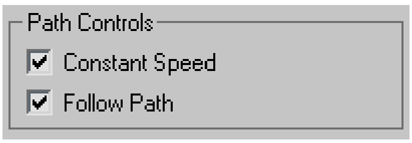
A new camera is created in the scene. The camera name, *Walkthrough_Cam01,* displays in the Cameras group.



3 In the Path Control group, click Pick Path and then in the Right viewport, click the blue spline named *Camera Path*.

The Pick Path button label changes to reflect the name of the selected spline. This constrains the Camera Position to the path. It also aligns it to follow the direction of the path.

NOTE By default, the camera will travel at a constant speed and will always point in the direction of travel. This is shown in the Advanced Controls rollout at the bottom of the dialog where both these options are enabled.



4 Right-click the lower-right Perspective viewport to activate it. In the Main Controls rollout, click the Set Viewport To Camera button to switch the viewport to the Camera view.

5  Click the Play Animation button to view the results in the Camera view.

The camera travels along the path but the motion seems unnatural, almost robotic. This is because the aim or target of the camera is controlled by the path constraint. It is far better to control the target manually, and choose the direction you want to look at any given time. As an analogy, as you walk in a straight line down a museum hallway, you would turn your head to look at the paintings on the wall instead of keeping your head pointed in the direction of your feet as you walk.

In the next lesson, you learn how to animate the camera using the Walkthrough Assistant.

Animating Camera Rotation

Now that the camera is constrained to the path, you'll animate the camera rotation so that it is aimed in a more natural position at points of interest in the scene.

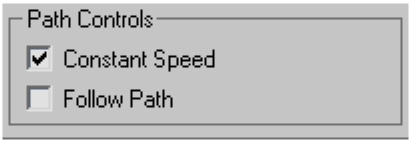
Set up the lesson:

- Continue from the previous lesson or on the Quick Access toolbar, click the Open File button and from the *\animation\walkthrough_asst* folder, open *great_wall_head.max*.

TIP If the Units Mismatch dialog displays, choose Adopt The File's Unit Scale and then click OK.
This is the same scene from the previous lesson. The camera is now in place but you will use Walkthrough Assistant to animate the Turn Head parameters.

Turning off Follow Path:

1. If the Walkthrough Assistant dialog is not visible, go to the Animation menu and choose Walkthrough Assistant.
2. In the Advanced Controls rollout, in the Path Controls group, disable the Follow Path option. You will not need it because you will control the head rotation manually.

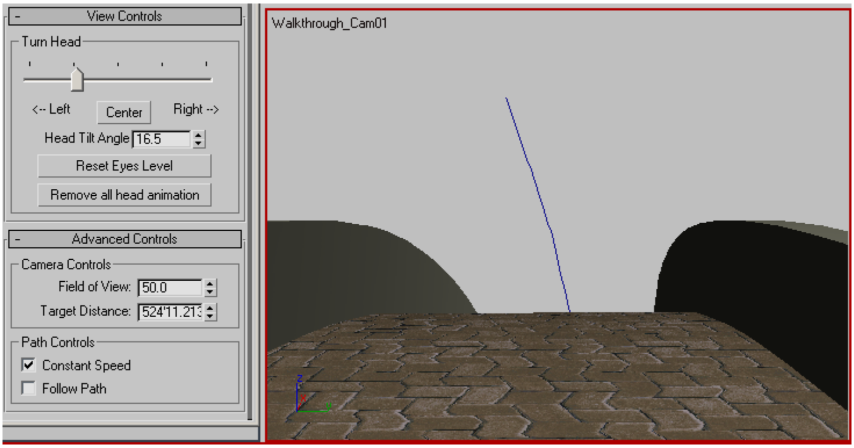


NOTE Disabling the Follow Path option resets the camera orientation to its default value (positive Y). You will fix that as you start animating the head rotation from the Walkthrough Assistant dialog.

3 

Make sure you are at frame 1, then turn on the Auto Key button.

4 In the View Controls rollout, move the turn head slider to the left and adjust the Head Tilt Angle to **16.4** in order get a better viewing angle of the brick path in the Camera viewport.



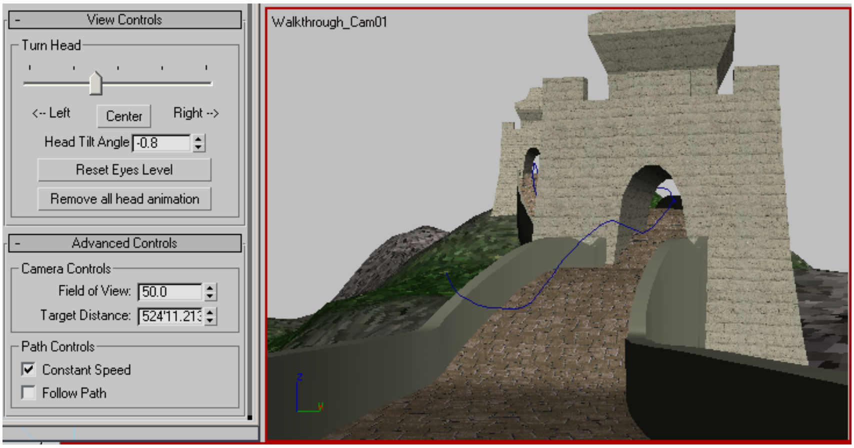
5 Scrub the time slider to frame 206. You are now at the other side of the hilltop. Adjust the Head Tilt Angle to approximately **-3.3** to level the camera head, then slide the Turn Head slider to the left to adjust the rotation.

The goal is to adjust the camera head rotation so that it looks towards the tower as if it has suddenly caught your attention.

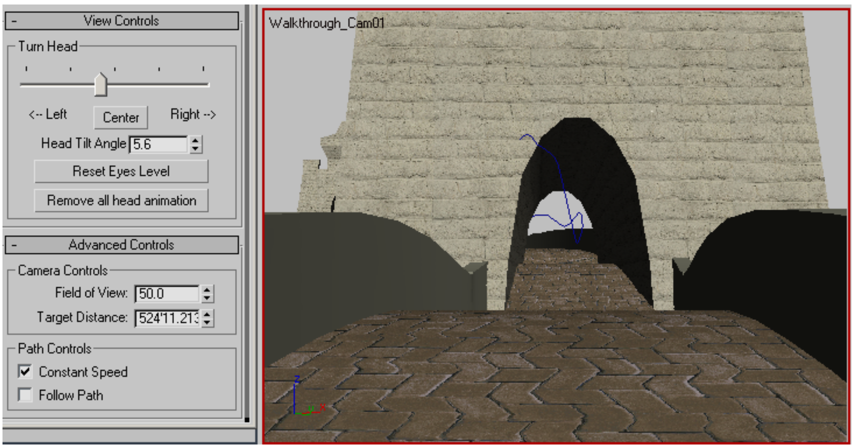
6 Scrub the time slider ahead to frame 408. Adjust the Head Tilt Angle to **–13.5** so that you are looking at the pavers in front of you.

Although it's nice to look at the scenery, it's also important to create a walkthrough that "feels' comfortable; in other words, to make it so that the viewer doesn't feel off balance while doing the a virtual walkthrough of your scene.

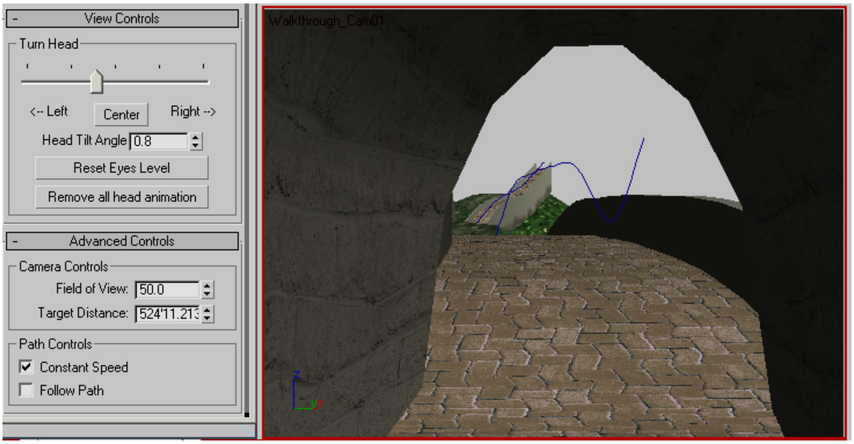
7 Scrub the time slider to frame 615. Change the Head Tilt Angle to **–0.8** and turn the Turn Head slider a little to the right until the camera is looking at the tower again.



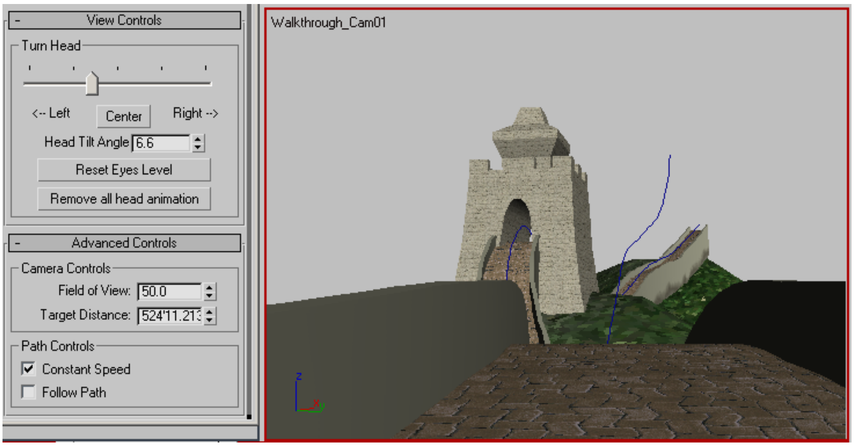
8 Scrub the time slider to frame 800. Change the Head Tilt Angle to about **5.6**. Move the Turn Head slider to the right until the tower opening is centered in the camera view.



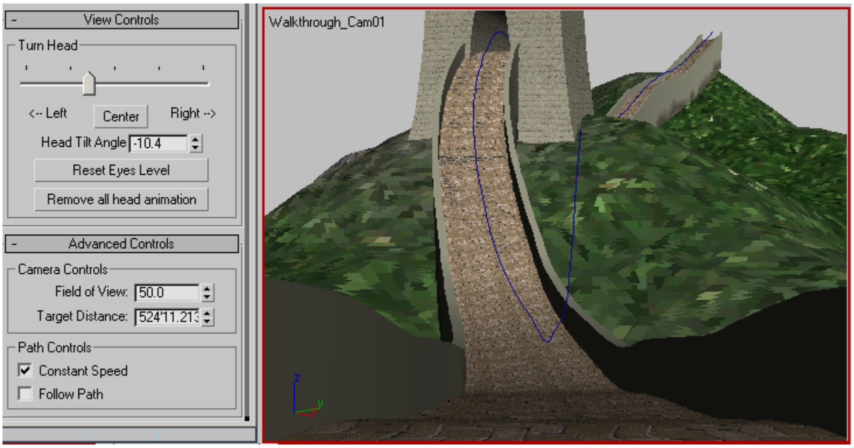
9 Scrub the time slider to frame 1050. Change the Head Tilt Angle to about **–0.8**. Move the Turn Head slider a little to the left so that the camera is tilted in anticipation of turning left after exiting the tower.



10 Scrub the time slider to frame 1150. Change the Head Tilt Angle to about **6.6**. Move the Turn Head slider slightly to the left so that you're looking the second tower in the distance.

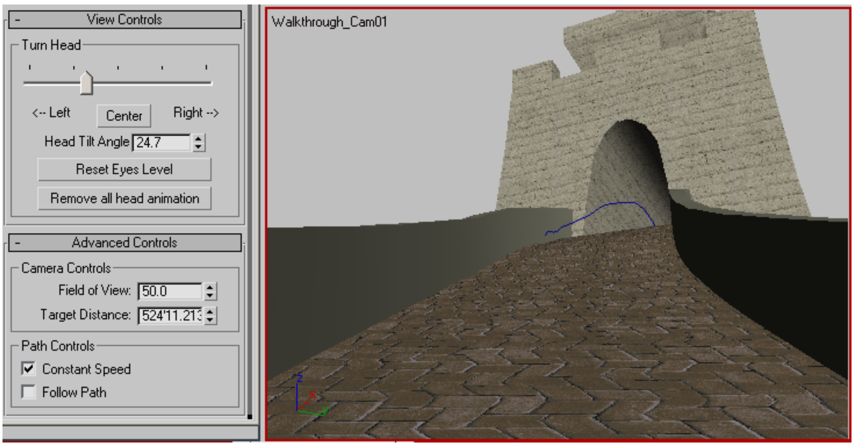


11 Scrub the time slider to frame 1280. This time change the Head Tilt Angle to about **–10.4** so that the camera head is tilted downwards to match the path's slope.



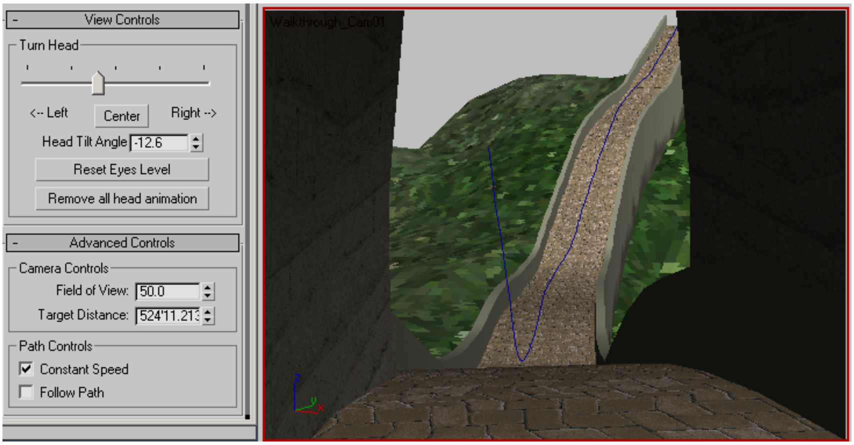
12 Scrub the time slider to frame 1420. Change the Head Tilt Angle to about **–11.9** to tilt the camera head further down.

13 Scrub the time slider to frame 1680. You're now looking up again towards the tower ahead of you. Adjust the Head Tilt Angle to about **24.7**. Slide the Turn Head slider a little to the left so that is aimed at the second tower.

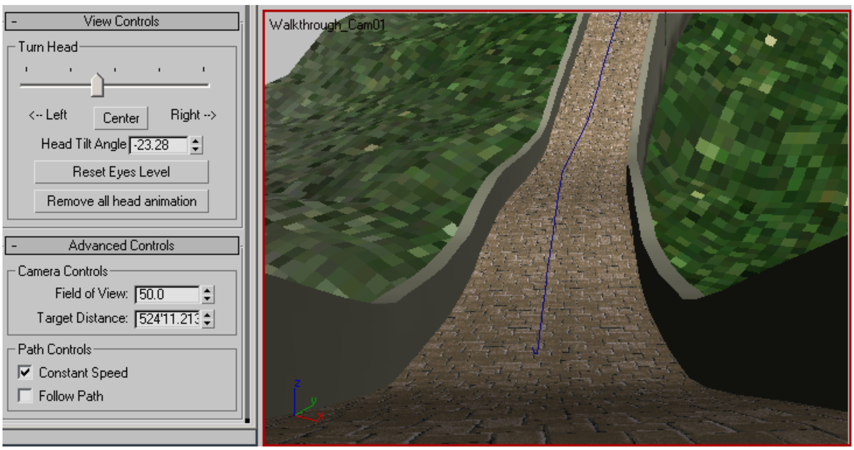


14 Scrub the time slider to frame 1860. Change the Head Tilt Angle to about **29.4**. Move the Turn Head slider slightly to the right so until you're looking at the second tower opening.

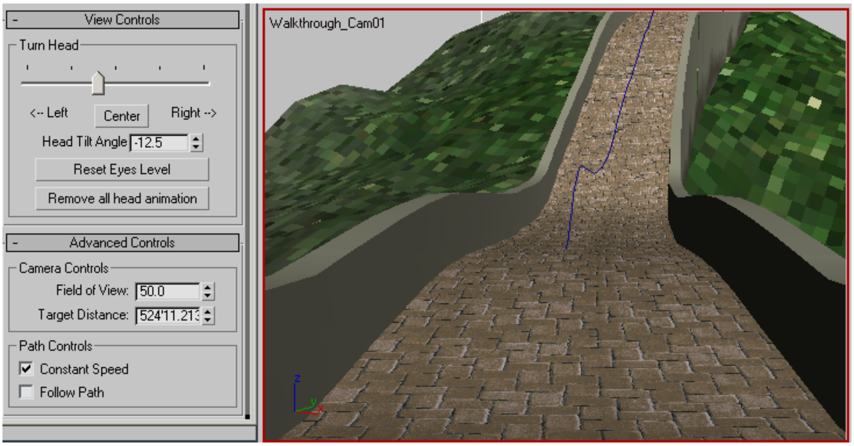
15 Scrub the time slider to frame 2030. Change the Head Tilt Angle to about **–12.6**.



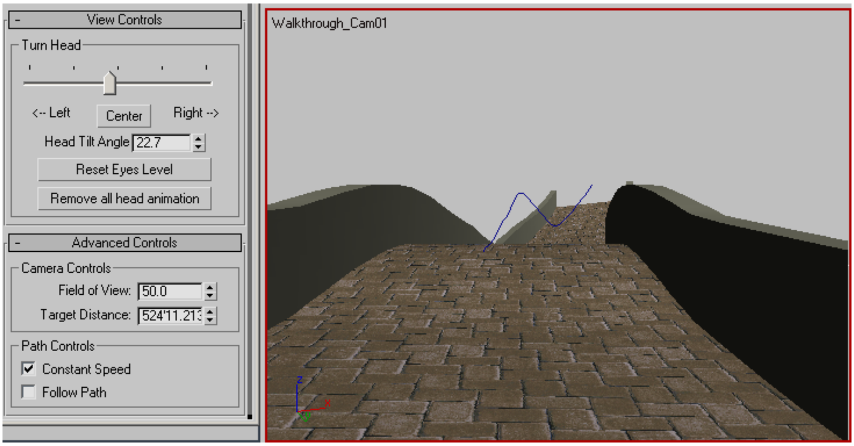
16 Scrub the time slider to frame 2125. At this point in the walkthrough, you are going down the hill again. Change the Head Tilt Angle to about **–23.3**. You want the camera head tilted downwards toward the path instead of the scenery because you want to feel as if you're watching your step.



17 Scrub the time slider to frame 2250. You are still looking at the path in front of you. Change the Head Tilt Angle to about **–12.5**. Adjust the Head turn and tilt to that purpose.



18 Scrub the time slider to frame 2550. You are now going up the path. Change the Head Tilt Angle to about **22.7**, and move the Turn Head slider slightly to the right so that you're looking towards the end of the pathway.



19 Turn off Auto Key mode.

20 Play the animation in the camera view to see the results. Notice that the camera motion is far more natural than it was in the previous lesson.

NOTE The camera path will not be visible when rendered.

---

21 Save your file as **great_wall_finish.max**.

Summary

In this lesson, you have created a camera walkthrough animation using the Walkthrough Assistant. You have learned to calculate the number of frames needed for the animation and you have learned to automate the creation of the camera and how to constrain it to a path. Finally, you have learned how to manually animate the head turn and tilt of the camera to create realistic camera motion.

Rendering Your Walkthrough Animation

There are some specific techniques to learn for rendering your animation into a movie file. You can render directly to a movie format such as AVI, or you can render a sequence of still image files to file formats such as TGA and then use the RAM Player to save them into a movie. The latter method is the recommended choice. It requires doing a few more steps than rendering directly to a movie format, but it gives you more control over the file size and quality of the output. In addition, if you have frames that artifacts or other errors, you can repair or remove them.

The next lesson will take some time to render. Depending on the speed of your computer, the rendering may take a few minutes to several hours.

Set up the lesson:

- From the *\animation\walkthrough_asst* folder, open *great_wall_render.max*.

  TIP If the Units Mismatch dialog displays, choose Adopt The File's Unit Scale and then click OK.

  This file is similar to the one created in the previous lesson. A bobbing motion has been added to the camera to simulate the up-and-down effect of someone jogging along the path. Two omni lights have been added to create additional lighting but there are no settings for Global Illumination in order to decrease rendering time.

Rendering an image sequence:

1. If the Camera viewport isn't active, right-click in it to activate it.
2. From the Rendering menu, choose Render Setup.

   Next, you'll define the animation range and output size.

3 On the Common tab of the Render Setup dialog, in the Time Output group, choose Range. Set the range of frames from **1** to **3000**.

4 In the Output Size group, change the output resolution to 320x240.

5 In the Render Output group, click the Files button.

The Render Output File dialog opens.

6 Navigate to a directory where there is enough disk space to save the rendered files. You can use the Create New Folder button to establish a new location, if necessary.

Next you'll define the type of still image file to render.

7 In the Save As Type field, click the drop-down arrow and choose JPEG File (*.jpg).

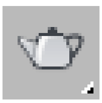
NOTE In a production environment, you may want to use a high-quality, lossless format such as TGA or TIF, but for the purposes of this tutorial, you will use the JPG format to keep the size of the output files small.

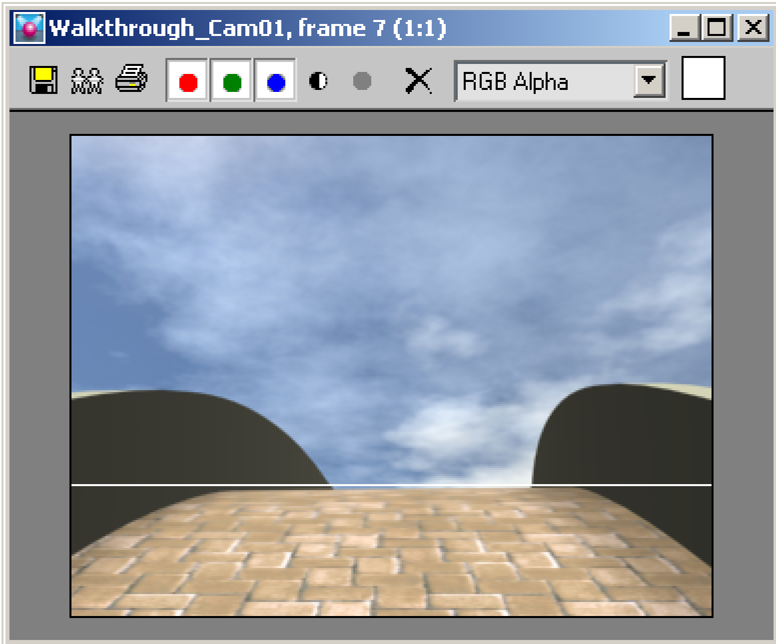
---

8 In the File name field, type **my_jog.jpg**, then click Save.

After you click Save, a format-specific dialog asks you to specify attribute and information settings. Accept the default values, and then click OK.

When you render a still-image sequence, as in this case, the software automatically appends the first part of the file name with a four-digit frame number. So the first frame will be *my_jog0000.jpg*, the second is named *my_jog0001.jpg*, and so on.

9  Make sure Save File is turned on in the Render Output group. Also check that the Viewport field at the bottom of the Render Scene dialog is set to Walkthrough_Cam01 (not Top, Front, or Left), then click Render Production.



NOTE A sky background is rendered. The dome representing the sky is an object that is hidden in the scene. However, the renderer is set to take hidden geometry into account.

10 The Rendering Progress dialog displays. Wait for a short while as the first frame is rendered. You will see the Last Frame Time, Elapsed Time, and Time Remaining values change after the first frame finishes.

Allow at least four frames to render.

At this point, you can work on something else while your animation is rendered.

TIP You can also watch the rendering for errors or observe to see where you want to make changes. This is generally a good practice so that you can study the scene as it is rendered.

After the rendering has completed, you will have 3000 JPG files in the folder you specified.

Convert an image sequence into a movie:

The RAM Player loads still image sequences into memory and plays them so you can watch them as a movie. It actually lets you load two different sequences and then compare them visually, but you won't use that functionality here. You'll simply use the RAM Player to save the files into an AVI file.

1 From the Rendering menu, choose RAM Player.

2 

On the RAM Player toolbar, click Open Channel A.

3 In the Open File Channel A dialog, navigate to the sequence of JPG image files. Highlight the name of the first file in the sequence and then ensure the Sequence option is turned on. Click Open.

The RAM Player will now load the image files in sequential order starting with the first file you selected. The Image File List dialog appears. Here you can use the Every Nth and Multiplier fields if you need to speed up or slow down your animation. If your animation is too slow, change Every Nth to 2 or 3. If your animation is too fast, increase the Multiplier.

4 Click OK.

The RAM Player Configuration dialog appears. Here you can observe and adjust your memory usage. There are also tools here to resize your animation, specify a range of frames to use, and split the alpha (transparency) information into a separate file.

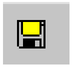
NOTE In order to use transparency, the image file specified has to be able to process an alpha channel. JPG files do not contain any transparency information the way TGA, TIF and PNG images often do.

5 Increase the Memory Usage to its maximum for your system, and then click OK

The RAM Player loads the rendered files into memory. In the Loading dialog, observe how much memory is being used and remains available.

If it looks like you are about to run out memory, click Stop Loading. If you have a low-memory system, reduce the number of frames to load and try again.

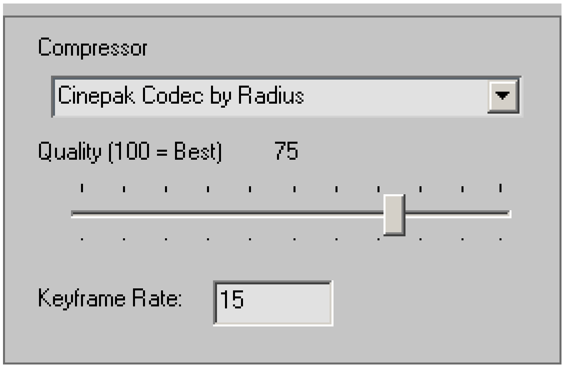
6 
On the RAM Player toolbar, click the Play button and watch the movie play.

7  On the RAM Player toolbar, click the Save Channel A button.

The Save File dialog appears.

8 Choose AVI as the file type, and name the animation **my_jog.avi**. Click Save.

The AVI File Compression Setup dialog appears. Here you can choose a codec (compression/decompression type) and adjust the quality of the file. Choose the default Cinepak Codec. To reduce file size, lower the quality to 75%.



9 Click OK to continue.

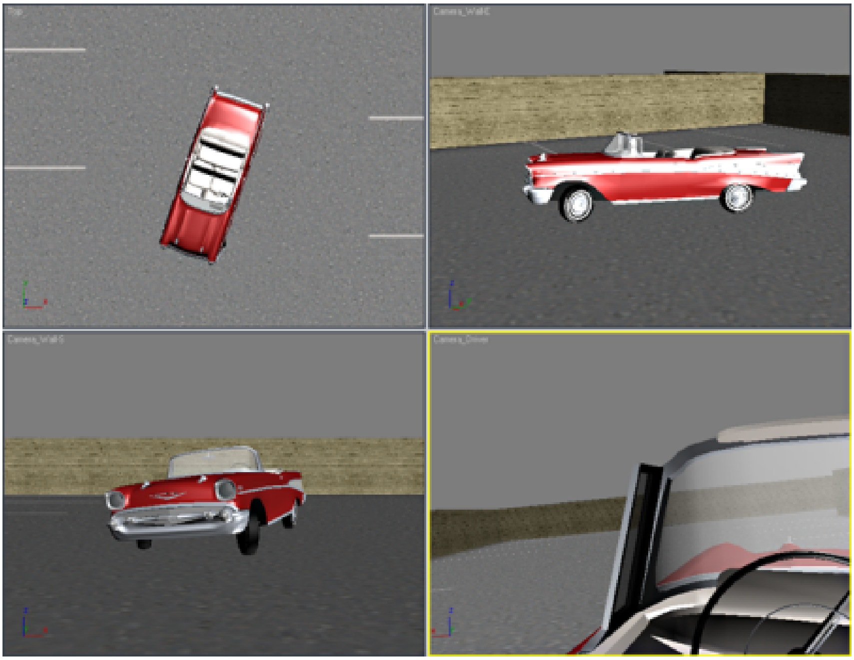
Summary

You have learned how to render your animation to a sequence of still image files. This allows you better control for later correction of your animation. You also learned how to assemble a still image sequence into a movie file, such as AVI or QuickTime using the RAM Player.

Rigging a Car

There is more to animating a car than simply giving it a trajectory. You should also consider such aspects as wheel rotation, the link between the steering wheel and front wheels, as well as body roll. Animating these aspects individually however, can become quite complicated.

In this tutorial you will establish relationships and other constraints to “rig” these moveable car parts so they can easily be animated together.



In this tutorial, you will learn how to:

- Use List controllers to manage animated components of a model
- Define controller behaviour though the use of expressions
- Use the MacroRecorder to automate the assignment of List controllers
- Create a toolbar to hold custom tools
- Use wiring and expressions to rig objects for animation

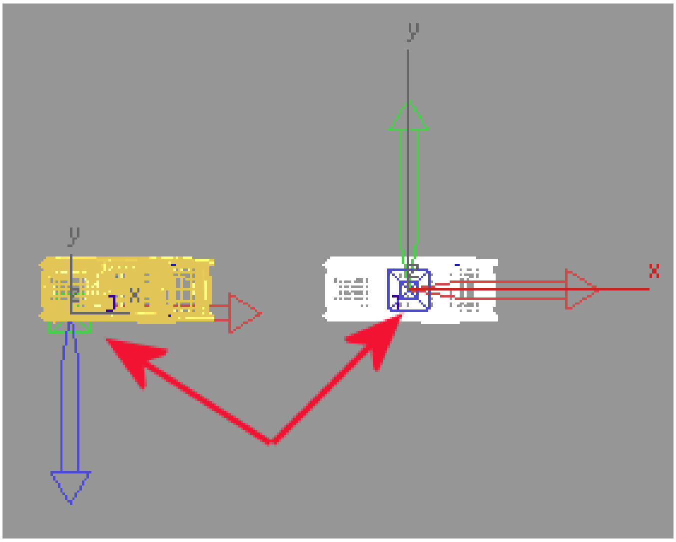
Skill level: Advanced

Time to complete: 1+ hours

Using List Controllers

A controller in 3ds Max is a plug-in that manages the values involved in keyframe animation, such as changes in object scaling, color, or translation. List controllers combine two or more controllers and can be very useful when combining relationships between objects.

List controllers, for example, are helpful when using expressions and constraints to control a child object through a parent object, particularly if the child and parent objects are not using the same orientation. The List controller uses added internal controllers that lets you maintain control over the child object's local orientation, even though it remains constrained to that of its parent.



Local orientation of child object (car wheel, shown at left in green) differs from parent object (car body, selected, shown at right)

The child object (car wheel) of the rig you are about to animate in this tutorial, is oriented differently from the parent object (the car body). To turn the wheel using wiring, you would have to rotate the wheel on its Y axis (based on the orientation of the body of the car), not its X axis (the wheel's local orientation). To regain control of the local orientation of the child object, you will add list controllers to the position and rotation tracks of the front left wheel animation.

Manually assign List controllers:

In this procedure, you will manually assign List controllers to the position and rotation tracks of the front left wheel of your Chevy.

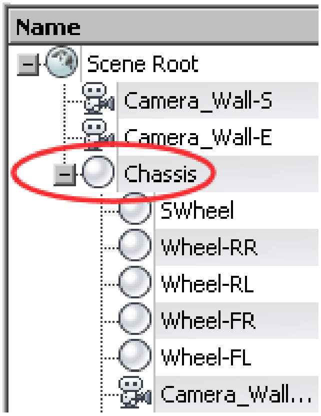
1  On the Quick Access toolbar, click the Open File button, navigate to the *animation\car rigging* folder and open *car_rig_01-start.max*.

2 From the main toolbar Selection Sets list, choose **Garage_All**.

3 Right-click the Perspective viewport and choose Hide Selection from the quad menu.

All scene objects other than the car are hidden.

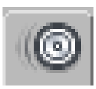
4 Press H to display the Select From Scene dialog and expand the *Chassis* object (the car body).



Notice how the four wheels, as well as the steering wheel, are children of the object . This hierarchy is typical to 3D car models.

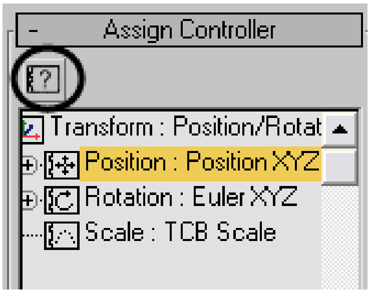
5 Choose Wheel-FL from the object list, then click OK.

The front left wheel in the scene is now selected.

6  Go to the Motion panel and expand the Assign Controller rollout.

A list of default animation controllers displays as tracks in an Explorer format.

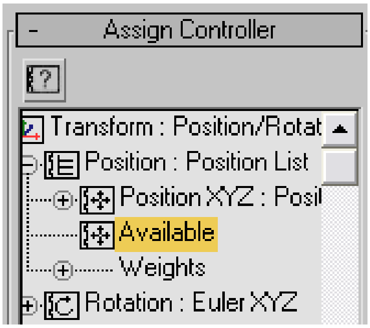
7 Highlight the Position:Position XYZ track, then click the Assign Controller button.



8 On the Assign Position Controller dialog, double-click Position List.

9 On the Assign Controller rollout > Position:Position List track, click the + icon to expand the position list.

10 Click the Available track, then click the Assign Controller button again.

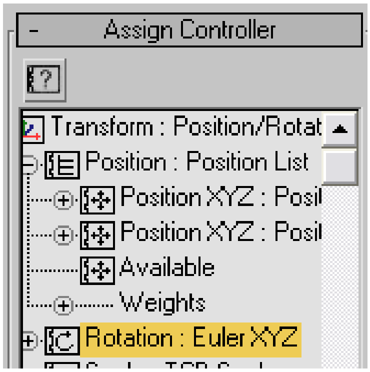


11 On the Assign Position Controller dialog, double-click Position XYZ.

In the explorer, a second Position XYZ: Position track has been created below the first. This track represents the controller that will control keyframe information of X, Y, and Z axes based on the local position of the child object (the front left wheel).

Next, you will repeat the procedure by assigning a List controller to the rotation track of the front left wheel.

12 On the Assign Controller rollout list of tracks, click Rotation: Euler XYZ and click the Assign Controller button.



13 On the Assign Rotation Controller dialog, double-click Rotation List.

14 On the Assign Controller rollout list of tracks, expand the Rotation: List track by clicking its + icon.

15 Click the Available track and click the Assign Controller button.

16 On the Assign Rotation Controller dialog, double-click Euler XYZ.

A second Euler XYZ track is created. This track controls rotation keyframe information of X, Y, and Z axes based on the local coordinates of the front left wheel.

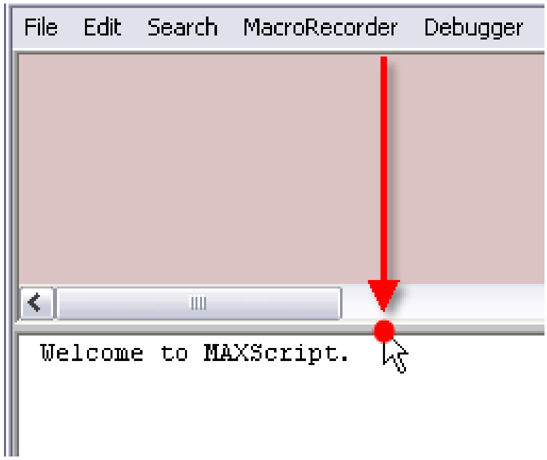
Automating the assignment of List controllers:

MacroRecorder is a simple scripting utility that records your interactions in 3ds Max. It converts your actions into a script that you can reuse to accomplish repetitive tasks.

In the previous procedure, you manually assigned position and rotation list controllers to a single wheel on your Chevy. You will now use MacroRecorder to automatically assign list controllers to the remaining three wheels, as well as the car body and steering wheel, so that the entire rig uses the same coordinate system in its animation.

1 Go to the bottom left corner of the interface, right-click anywhere on the MAXScript area and choose Open Listener Window.

2 Directly below the MAXScript Listener menu bar, click and drag downward to reveal the pink MacroRecorder panel, if it is not already visible.

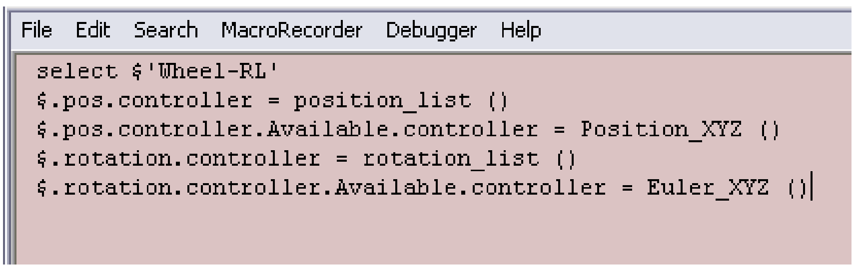


3 From the menu bar, choose MacroRecorder > Enable.

From this point onward, virtually any action you take in 3ds Max will be recorded in a script.

4 Select the rear left wheel of the car (the *Wheel-RL* object).

5 Repeat steps 6 through 16 of the previous procedure to assign list controllers to the position and rotation of the rear left wheel.



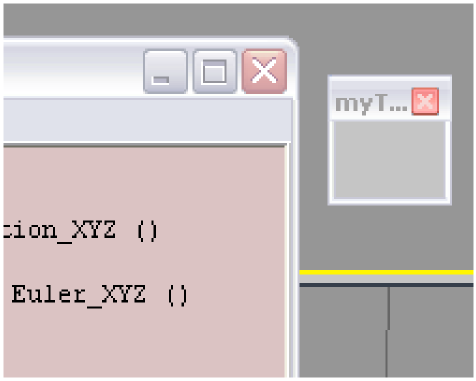
As you progress, note how the pink MacroRecorder area accumulates scripting data.

6 Right-click on a gray area of the main toolbar (below the Selection Sets drop-down is a handy area), then choose Customize.

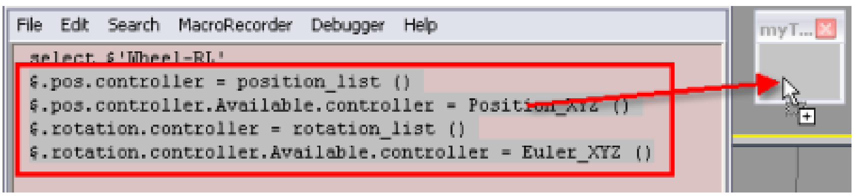
7 On the Customize User Interface dialog, make sure the Toolbars tab is active, then click New.

8 On the New Toolbar dialog, type **myTools** and click OK.

9 Close the Customer User Interface dialog and reposition the new toolbar to the right of the MAXScript Listener window.



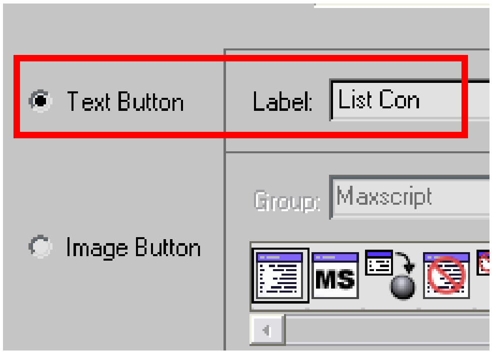
10 Highlight the last four lines of the script, then drag and drop them into the myTools toolbar.



A button is created.

11 Right-click the newly-created button and choose Edit Button Appearance.

12 On the Edit Macro Button dialog, choose the Text Button option and in the Label field, type **List Con** and click OK.



13 On the MAXScript Listener window menu bar, choose MacroRecorder > Enable to turn off script recording.

The MacroRecorder stops recording your interactions in 3ds Max.

14 Close the MAXScript Listener window, then resize the myTools toolbar until the List Con label is fully displayed.

You are now ready to use the List Con tool to quickly assign list controllers to the remaining wheels of your car model.

15 In any viewport, select the front right wheel of the car (the *Wheel-FR* object).

16 On the myTools toolbar, click List Con.

On the Motion panel > Animation Controller rollout, expand the Position XYZ track to display one of the list controllers that was assigned by the script you just created.

17 Select the last remaining wheel in the model and click List Con again.

18 Repeat the previous step for the *Chassis* object.

19 Repeat the previous step for the *SWheel* object.

NOTE You can apply the MacroRecorder script to only one object at a time. You must therefore click the List Con button once for each object you want to modify.

---

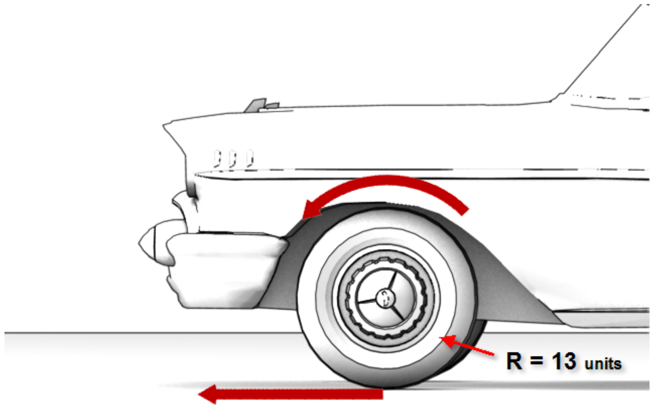
20 Save your work as **mycar_rig_02.max**.

The myTools toolbar you created in this procedure is now available for all future 3ds Max work sessions.

In the next lesson, you will learn how to animate the rotation of the car wheels.

Rotating the Wheels

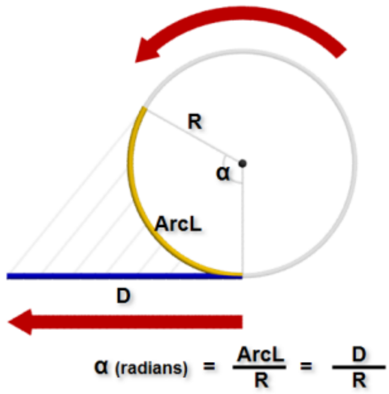
In this lesson, you will learn how to rotate the wheels by an amount that corresponds to the distance travelled by the car model.



Let's start by taking a look at the trigonometry involved in calculating the wheel rotation.

In any circular object, the amount of rotation (α) is defined by the radius of the circle and the arc length encompassed by the angle. That amount of rotation (α) expressed in radians is equal to the arc length, divided by the radius of the circle (arc length / R), where:

- the **radius** of the car wheel is constant and equal in this case to 13 units.
- the **arc length**, when flattened, represents the distance travelled by the car and its wheels.



Therefore, the wheel rotation calculation (arc length / R) becomes **distance / 13**. Whereas the radius of the wheel is constant and equal to 13, the distance travelled is variable.

Set up the lesson:

- Continue from the previous lesson or open *car_rig_02.max*.

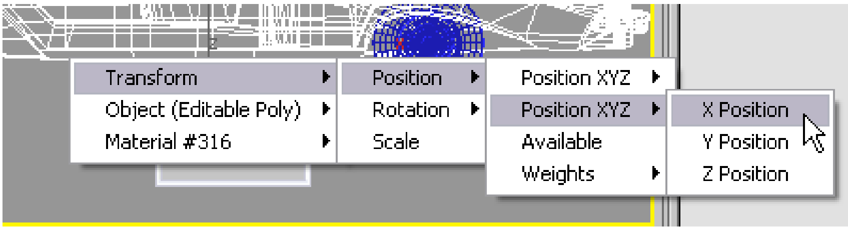
Rotate the wheels (in World X coordinates):

1. In the Perspective viewport, select the car body.

   The car is currently oriented on the World X axis: you will begin working in this coordinate system.

2. Right-click the car body object and from the quad menu, choose Wire Parameters.

3. From the menu, choose Transform > Position > (2nd) Position XYZ > X Position.

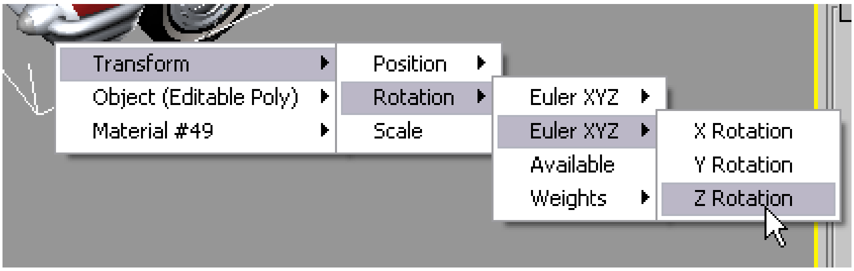


NOTE It is important to always leave the first animation controller at the top of the list (in this case, the Position XYZ Controller) untouched, since it serves as a "lock" for the parent/child relationship. When choosing controllers to work on, always work from top of the controller list downward, starting with the second controller.

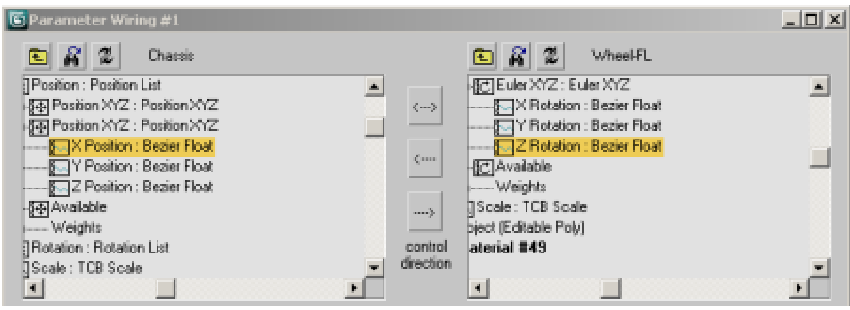
A rubber band shows the link you are about to make between your two selected objects.

4 Select the front left wheel of the car (*Wheel-FL*).

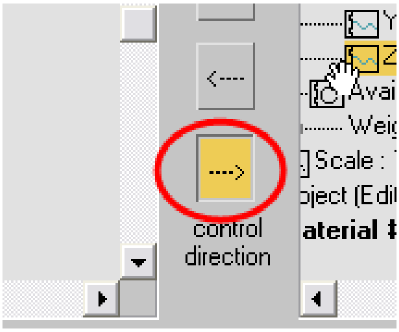
5 From the menu, choose Transform > Rotation > (2nd) Euler XYZ > Z Rotation.



The Parameter Wiring #1 dialog opens. You use this dialog to set up one and two-way control relationships between objects. The position and rotation of the two objects you just selected to affect one another are highlighted.



6 On the Parameter Wiring dialog, click the right-pointing arrow above "control direction".

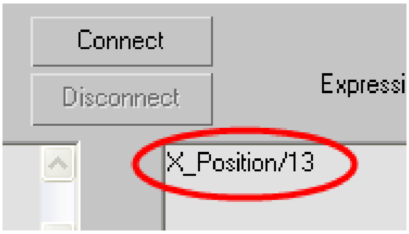


This ensures that the *Chassis* X position is controlling the *Wheel-FL* Z rotation and not the other way around.

The bottom-right corner of the Parameter Wiring dialog displays the wheel object Expressions panel. It shows the distance travelled as X_Position.

7 Next to X_Position, type **/13**.

The expression should now read X_Position/13, the distance divided by the radius of the wheel.



8 Click Connect, but do not close the dialog.

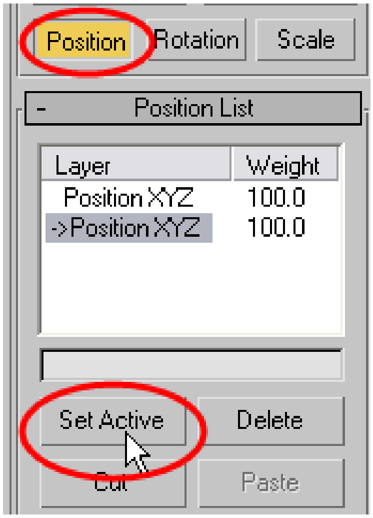
9 Test your work by moving the car body on its X axis.

Note how the front left wheel does not rotate. Even though you added a position list controller to the car and wheel, the first controller in the list (the one that ensures the parent/child "lock") is still active. You need to make the second position controller (the one used in the wiring process) the active one.

10 If you moved the car model, press Ctrl+Z to undo the move.

11 With the car selected, on the Motion panel > PRS Parameters rollout, click the Position button at the bottom of the rollout.

12 On the Position List rollout, highlight the second Position XYZ controller and click Set Active.



13 Try moving the car on its X axis again.

TIP To better see the wheel rotation, you can switch your viewport display to Smooth + Highlights.

The wheels now rotate and at the correct rate, but they do so in a backward direction.

14 On the Expressions panel, add a minus (-) in front of the expression and click Update.

15 Move the car on its X axis again and note how the wheel rotates in the proper direction.

16 Repeat the preceding steps for each of the remaining three car wheels.

Because the wheels were mirrored, the wheels on the right side of the car do not need the minus sign added to their expression, whereas those on the left side do.

17 Close all the Parameter Wiring dialogs.

Add subcontrollers for Y rotation:

In the previous procedure, you learned how to add controllers that determine car wheel rotation for the length of distance travelled by the model along the World X axis. However, if you tried to rotate the car in any way, wheel rotation would be reduced or stop altogether. You therefore need to add controllers that account for the car's displacement in a Y direction.

1. In the Top viewport, select the car body object and rotate it 90 degrees clockwise so that its front bumper points at 12 o'clock.

   The car is now oriented on the World Y axis, so you will begin working in this coordinate system.

2. If required, adjust the view in the Perspective viewport until you can see the front left side of the car.

3. Move the car forward and backward on the Y axis. Note that the wheels do not rotate.

   To get the wheels rotating, you will need additional animation controllers, ones that will control the car's displacement in the Y direction. You will add these as sub-controllers, so you do not overwrite the controllers already in place.

4. Go to the bottom-left corner of the interface, right-click the MAXScript area and click Open Listener Window.

5. On the MacroRecorder panel, highlight the line that reads:

   $.rotation.controller.Available.controller = Euler_XYZ ()

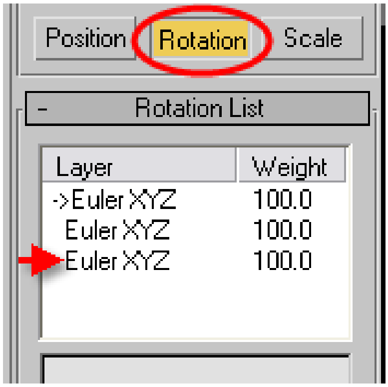
   Be sure not to include the line's carriage return when you make your selection.

6. Press Ctrl+C to copy this line to memory. If you are not continuing from the previous lesson, this line will not be available from the Open Listener window. If this is the case, copy the line from this .pdf document.

7. Close the MAXScript Listener window, then select the front left wheel (*Wheel-FL*).

8. On the bottom-left corner of the interface, click inside the white entry box, press Ctrl+V to paste the line of code, then press Enter.

9. On the Motion panel > PRS Parameters rollout, make sure that the Rotation button is active verify that a new sub-controller has been added to the rotation list.



10 Repeat step 8 to add a fourth rotation sub-controller. You will need this later on in the tutorial.

The front left wheel should now have four Euler XYZ tracks.

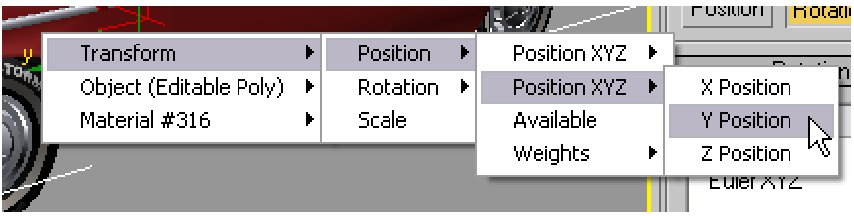
11 Select another wheel and repeat steps 8 to 10 until all four wheels have four Euler XYZ tracks in their respective rotation lists.

Rotate the wheels (in World Y coordinates):

1 Adjust the Perspective view until the front left side of the car is visible.

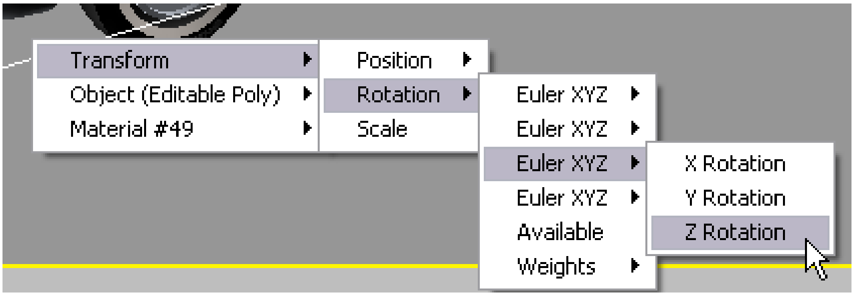
2 Select the car body, then right-click and from the quad menu, choose Wire Parameters.

3 From the menu, choose Transform > Position > (2nd) Position XYZ > Y Position.



4 Select the front left wheel (*Wheel-FL*).

5 From the menu, choose Transform > Rotation > (3rd) Euler XYZ > Z Rotation.



6 On the Parameter Wiring dialog, click the right-pointing arrow above Control Direction to ensure that the *Chassis* Y position is controlling the *Wheel-FL* Z rotation.

7 On the right-hand Expressions panel, type **/13**.

The expression for the left-hand wheel should be **Y_Position/13**

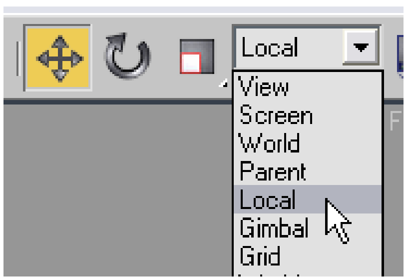
8 Click Connect.

9 Repeat steps 3 to 8 for each of the other three wheels.

NOTE The expression for the right-hand wheels should be **-Y_Position/13**.

10 Close the Parameter Wiring dialogs and in the Top viewport, rotate the car so that it is not pointing horizontally or vertically.

11 On the main toolbar, click Select And Move, then set the coordinate system to Local.



12 Adjust the Perspective viewport, so you can see the car from its side.

13 Move the car on its local X axis. Note how the wheels are rotating properly.

14 In the Top viewport, rotate the car until the front bumper is pointing to the left.

15 Save your file as **mycar_rig_03.max**.

Rotate the wheels (under a path constraint):

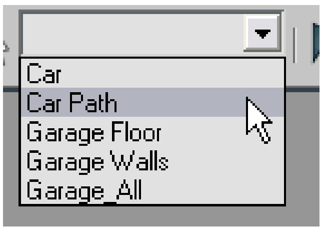
In the previous procedure, you learned how to add controllers that rotate the car wheels for any distance of travel in World X and Y space. The wheels will therefore rotate properly when you manually move the car around the scene in any direction.

However, you would most often animate motion of a car by placing it on a pre-defined path using Path Constraint. This type of animation requires a different expression.

This new expression uses the same formula (distance divided by radius) as the ones you have been using, but while the radius of the wheel remains constant, the distance travelled is calculated differently.

1 Continue from the last procedure or open the file *car_rig_03.max*.

2 From the main menu Selection Sets list, choose Car Path.

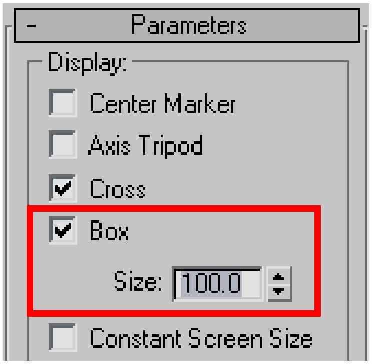


A warning message displays.

3 Click Yes to display the path you will use to animate the car motion.

4 From the main menu, choose Create > Helpers > Point.

5 On the Parameters rollout, turn on Box and set Size to **100.0**.

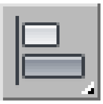


This increases the size of the helper gizmo and makes it easier to select in the scene.

NOTE Many animators use the Dummy helper instead of Point. The advantage of using a Point helper is you can adjust its size without having to scale it. Scaling a helper in a hierarchy will affect its children objects, something you usually want to avoid.

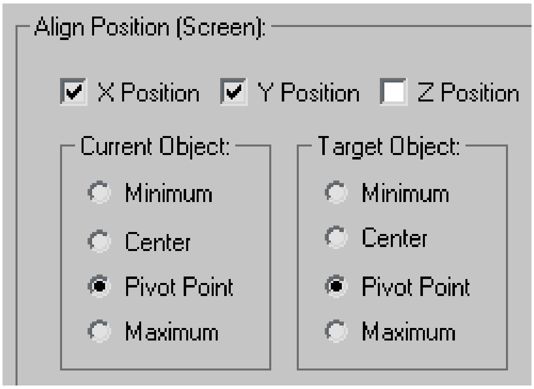
---

6 In the Top viewport, click a point near the car to place a Point helper.

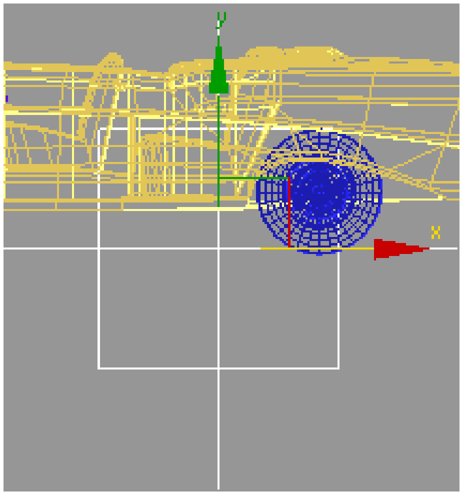
7  With the Point helper still active, on the main toolbar click Align, then in any viewport, select the car body.

8 In the Align Selection dialog > Align Position group, make sure X Position and Y Position are on and Z position is off.

9 In the Current Object and Target Object groups, choose Pivot Point, then click OK.



10 In the Front viewport, move the Point helper on its X axis to the right until it is just to the left of the rear axle of the car.

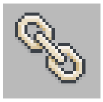


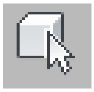
Point helper to left of rear axle

The Point helper location you specify becomes the pivot point of the car when the front wheels turn.

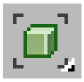
11 On the command panel > Name And Color rollout, rename the helper **Dummy_CAR**.

12 In any viewport, select the car body.

13  On the main toolbar, click Select And Link, then in the Front viewport, click the car body and drag to the Point helper. This makes the car body the child of the Point helper.

14  On the main toolbar, click Select Object to exit link mode.

15 From the main toolbar Selection Sets list, choose *Garage_All*. Click Yes to dismiss the warning and unhide the rest of the scene geometry.

16  In the Top viewport, use Zoom Extents to view the entire parking lot.

17 In the Perspective viewport, click the Perspective label and from the menu, choose Cameras > Camera_Wall-E.

Animate the dummy by constraining it to a path:

1 In any viewport, select the *Dummy_CAR* helper.

2 From the main menu, choose Animation > Constraints > Path Constraint.

3 In the Top viewport, click on the green path (*CarPath*).

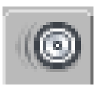
The helper and the linked car are repositioned at the start of the path.

NOTE You could, as an alternative, constrain the car directly to the path. In this case, however, it is preferable to constrain the helper parented to the car so you can retain extra control over the car's behavior (such as defining skids around tight corners).

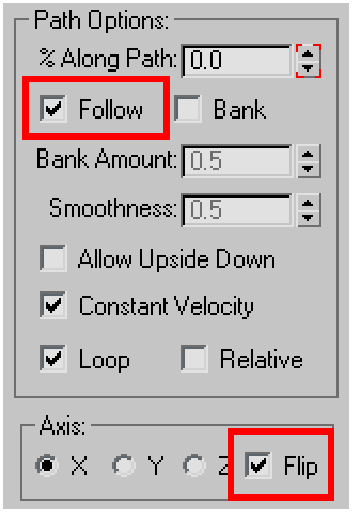
---

4 Scrub the animation.

The car's orientation remains constant throughout the animation.

5  In the Motion panel > Path Parameter rollout > Path Options group, turn on Follow.

6 Scroll down to display the Axis group and turn on Flip.



The Flip option prevents the car from driving in reverse.

7 Scrub the animation again.

Car motion is improved, but at the last frame the car points at an awkward angle. This is a common behavior to paths based on a NURBS curve. You will now correct this problem.

NOTE NURBS curves, when used as animation paths, provide a smoother "ride" than regular splines.

---

8 Go to the last frame of the animation (frame 150), and make sure the Point helper is selected.

9 

Turn on Auto Key mode.

10 In the Motion panel > Path Parameters rollout > Path Options group > % Along Path box, type **99.9** and press Enter.

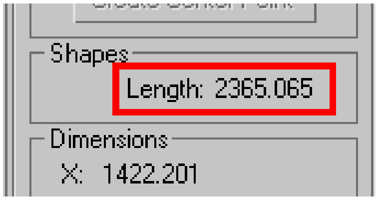
11 Turn off Auto Key and scrub the animation.

The car is properly oriented on the path, but the wheels no longer rotate. This is because the expression that defined the wheel rotation you formulated earlier no longer applies. The distance travelled by the car was dependent on the X and Y displacement in the World coordinate system. Displacement is now tied to the length of the path and the percentage of the path that the car has travelled. You must therefore modify the expression to reflect this change.

Wire wheel rotation to a path:

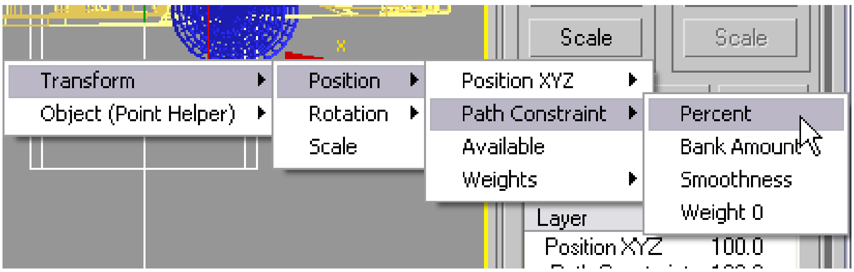
1  In any viewport, select the animation path (*CarPath*) then go to the Utilities panel.

2 Click Measure and in the Shapes group, take note of the path length.

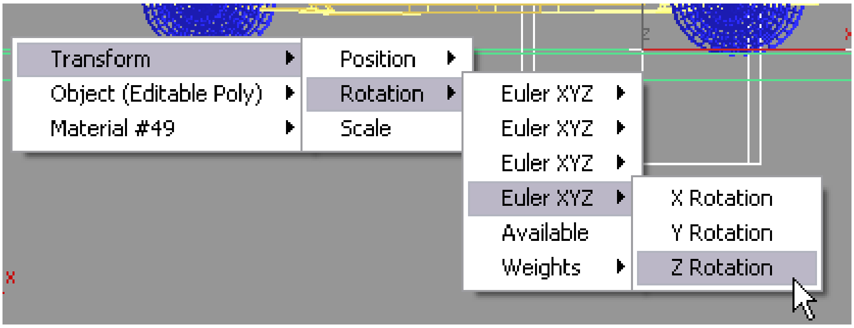


3 Select and right-click the Point helper, then from the menu choose Wire Parameters. (You may need to adjust the model in the Perspective viewport to better select the helper.)

4 From the menu, choose Transform > Position > Path Constraint > Percent.



5 Click one of the car wheels and choose Transform > Rotation > 4th Euler Rotation > Z Rotation.



6 On the Parameter Wiring dialog, set the control direction to the right, which places the Percent parameter in control of the wheel rotation.

7 On the right-hand Expressions panel, type **(2365*Percent)/13**.

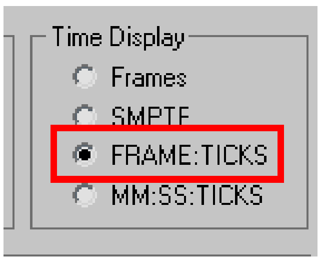
NOTE The value 2365 is the length of the animation path you measured earlier. When multiplied by the percent variable, it calculates the distance the car has travelled at any given moment in time along the path. When divided by the radius of the wheel (13), it provides the amount of rotation needed for the wheel to turn.

---

8 Click Connect.

9 Scrub the animation to see the wheel rotation.

10  To better see the animation, click the Time Configuration button and in the Time Configuration dialog > Time Display group, turn on FRAME:TICKS.



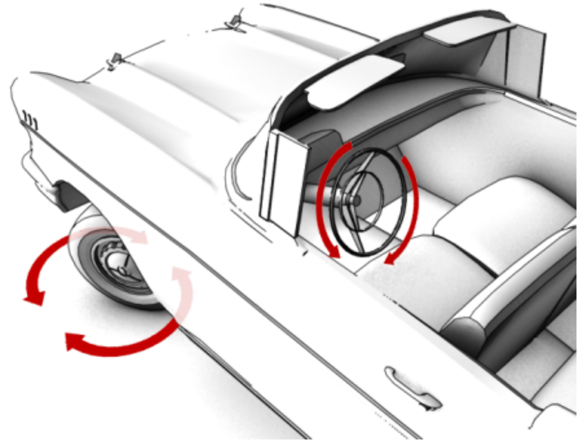
11 Repeat steps 3 to 8 to link the Point helper to each of the remaining three car wheels.

Remember to add a minus (-) operator to the expression of the wheels on the right side of the model so they don't rotate in the opposite direction.

12 Save your file as **mycar_rig_04.max**.

Pivoting the Wheels

You now need to make sure the front wheels pivot or "turn" as the car moves left or right along the animation path. For added realism, you will also establish a relationship between the wheel pivot and the turn of the steering wheel.



Set up the lesson:

- Continue from the previous lesson or open *car_rig_04.max*.

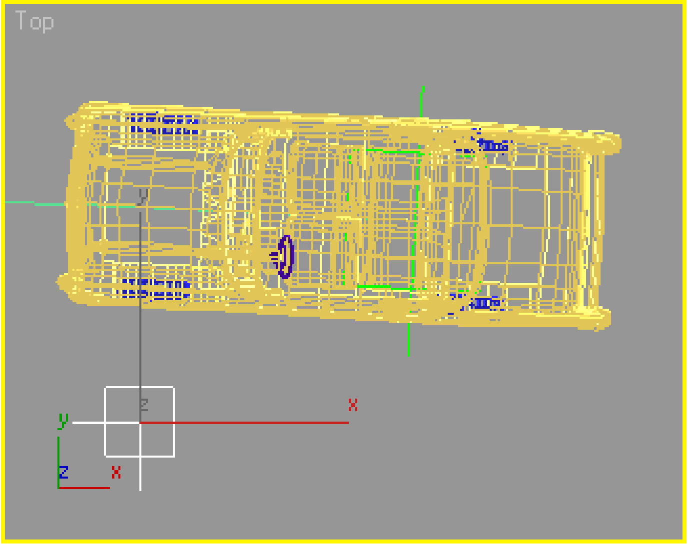
Set up helpers for the front wheels:

In the same way you created a Point helper to direct the car animation along a path, you will also create two more Point helpers to control the pivot of the front wheels by the rotation of the steering wheel.

1 In the Top viewport, zoom in on the car and press F3 to switch to Wireframe mode.

2 From the main menu, choose Create > Helpers > Point.

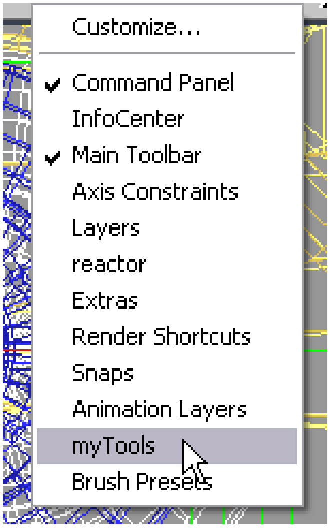
3 Click anywhere around the car body and in the Parameters rollout, turn on Box, then in the Size box, type **50.0** and name the helper **Dummy_FL**.

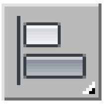


4 With the helper still selected, click the List Con button on the myTools toolbar you created earlier in the tutorial.

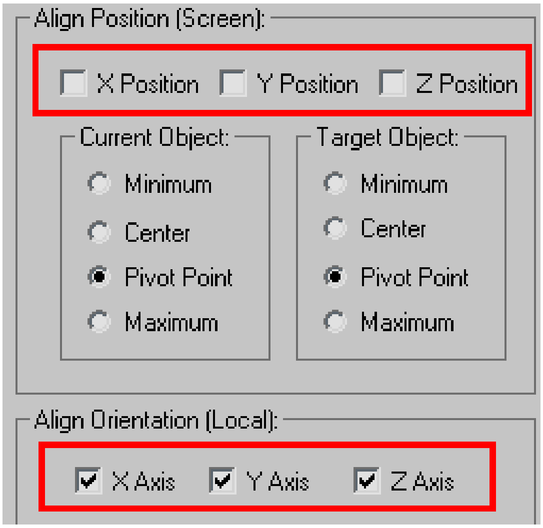
The List Con script automatically assigns the two Position list and Rotation list controllers you set up earlier, permitting you to retain control over the helper's local orientation.

NOTE If the myTools toolbar is not currently displayed on your interface, right-click a gray area on the main toolbar and choose myTools from the menu.



5  From the main toolbar, click Align and in the Top viewport, select the car body.

6 In the Align Selection dialog > Align Position group, turn off X Position, Y, Position and Z Position. In the Align Orientation group, turn on X Axis, Y Axis and Z Axis.



These settings ensure that the car and the helper have the same orientation.

7 Click OK to close the dialog.

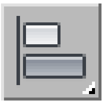
8 From the main toolbar, use Shift+Move and drag the helper to make a copy.

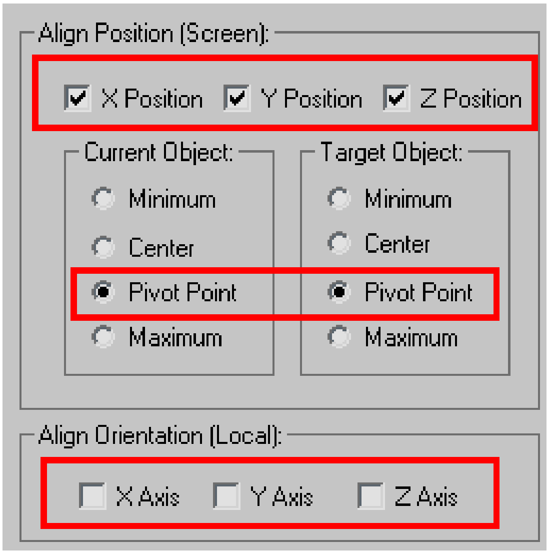
TIP Set the coordinate system to Local to make moving the Point helper easier.

---

9 In the Clone Options dialog, name the copy **Dummy_FR**.

Next, you will align the helper and the right front wheel pivot to pivot in X, Y and Z positions.

10  In the Top viewport, select the Chassis object, right-click and choose Hide Selection. Click the Align tool then zoom in and click *Wheel_FR*.



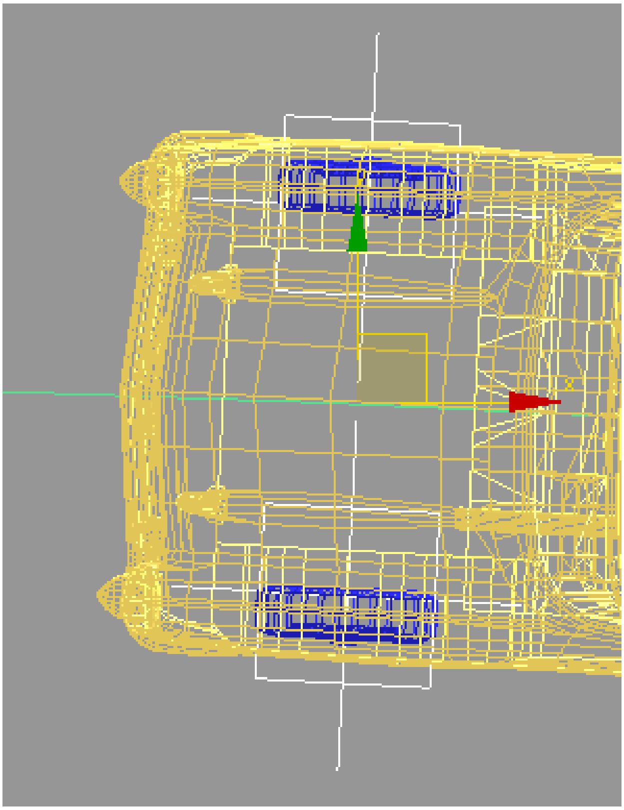
11 In the Align Selection dialog > Align Position group, turn on X Position, Y Position, and Z Position and choose Pivot Point for both the Current Object and Target Object.

12 In the Align Orientation group, turn off X Axis, Y Axis and Z Axis.

13 Click Apply, then OK to exit the dialog.

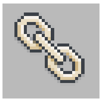
14 Select *Dummy_FL* and from the main toolbar click the Align tool. In the top viewport click *Wheel_FL*.

15 Repeat steps 11 to 13 to align the helper to the front left wheel.



Point helpers aligned to front left and right wheels

You will now rework the hierarchy and parent/child relationships of the car setup to prepare for the body roll you will rig in the next lesson.

16  On the main toolbar, click Select And Link.

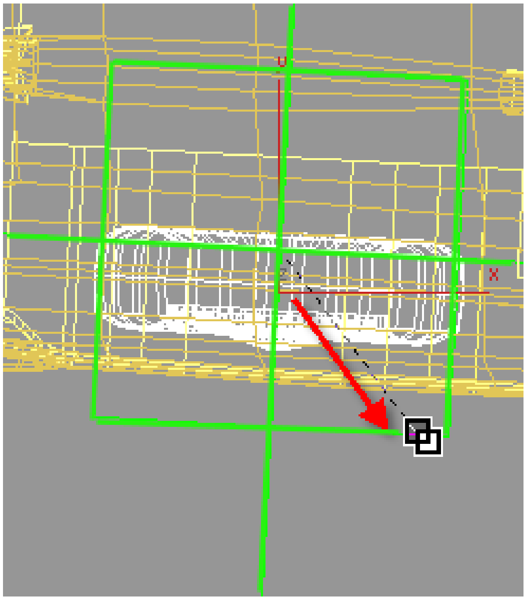
Link the wheel helpers to the car helper:

1 Ctrl+select the two wheel helper objects, then drag to the *Dummy_CAR* object.

This links the helpers as children of the *Dummy_CAR* object.

2 With the Select and Link tool still active, select both rear wheels (*Wheel-RL* and *Wheel-RR*) and drag to the *Dummy_CAR* helper.

3 Select the front left wheel (*Wheel-FL*) and link it to the *Dummy_FL* helper.



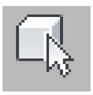
Linking front left wheel to front left wheel helper

4 Select the front right wheel (*Wheel-FR*) and link it to the *Dummy_FR* helper.

5 In the viewport, right-click and choose Unhide By Name. On the Select From Scene dialog, choose Chassis.

Rotate the wheels (in World X coordinates):

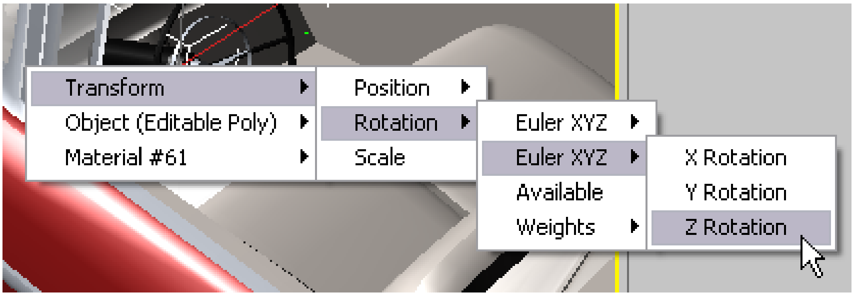
1 Click the Camera_Wall-E viewport label and from the menu, choose Cameras > Camera_Birdseye.

2  On the main toolbar, click the Select tool.

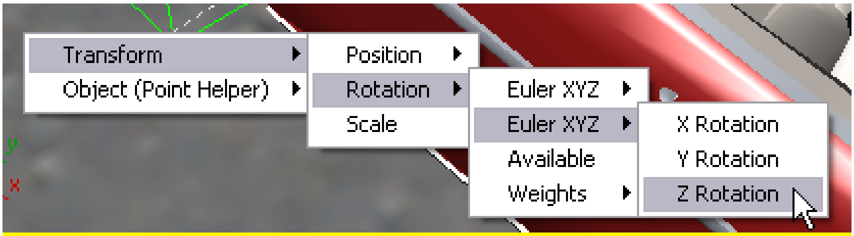
3 In the camera viewport, select the steering wheel (*SWheel*).

4 Right-click the steering wheel and choose Wire Parameters.

5 From the menu, choose Transform > Rotation > (2nd) Euler XYZ > Z Rotation.



6 Select the *Dummy-FL* object, which is the front left wheel helper, and choose Transform > Rotation > (2nd) Euler XYZ > Z Rotation.



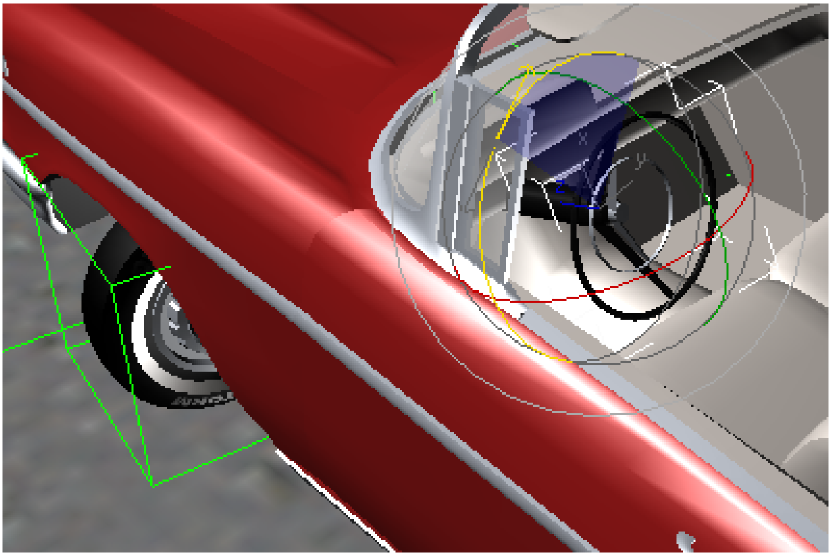
7 On the Parameter Wiring dialog, set the control direction to both ways, since the manual turning of either object affects the other.

8 Click Connect and leave the Parameter Wiring dialog open.

9 On the main toolbar, click Select And Rotate and set the coordinate system to local.

10 Rotate the steering wheel on its local Z axis.

Note how the steering wheel and the front wheel turn in the opposite direction. You will correct this by modifying the controller expression. A second adjustment is also required. The front wheel needs to turn far less than the steering wheel. This is because the pivot range of a front wheel is about 90 degrees, whereas a steering wheel range of movement is two to three complete revolutions.

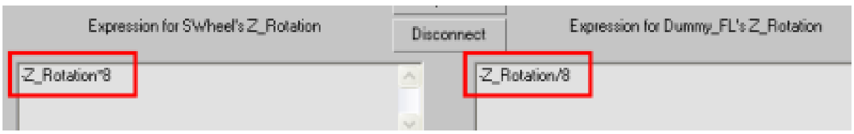


Steering wheel and front wheel turn in opposite directions

11 Cancel or undo the rotation you made in the previous step.

12 In the left-hand Expressions panel, under “Expression for SWheel’s Z_Rotation”, type: **-Z_Rotation*8**.

13 In the right panel, under “Expression for Dummy_FL’s Z_Rotation”, type: **-Z_Rotation/8**.



The minus (-) operator ensures that the two rotations are aligned, and the *8 and /8 factors ensure that the front left wheel pivots (rotates in Z) eight times less than the rotation of the steering wheel.

14 Click Update and test your work by rotating the steering wheel on its local Z axis again.

Note the more realistic behavior.

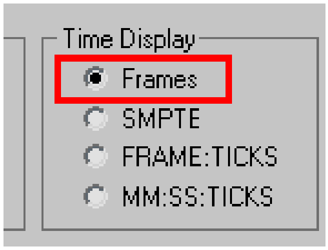
15 Repeat steps 4 to 14 to wire the steering wheel and the front right wheel helper.

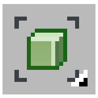
Be sure to specify the exact same expressions as you did in steps 13 and 14, since the right wheel helper was copied, not mirrored, from the front left helper.

16 Close the Parameter Wiring dialogs.

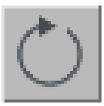
Animate the steering:

1  If your timeline is displayed in frames and ticks, click the Time Configuration button and in the Time Configuration dialog > Time Display group, choose Frames.



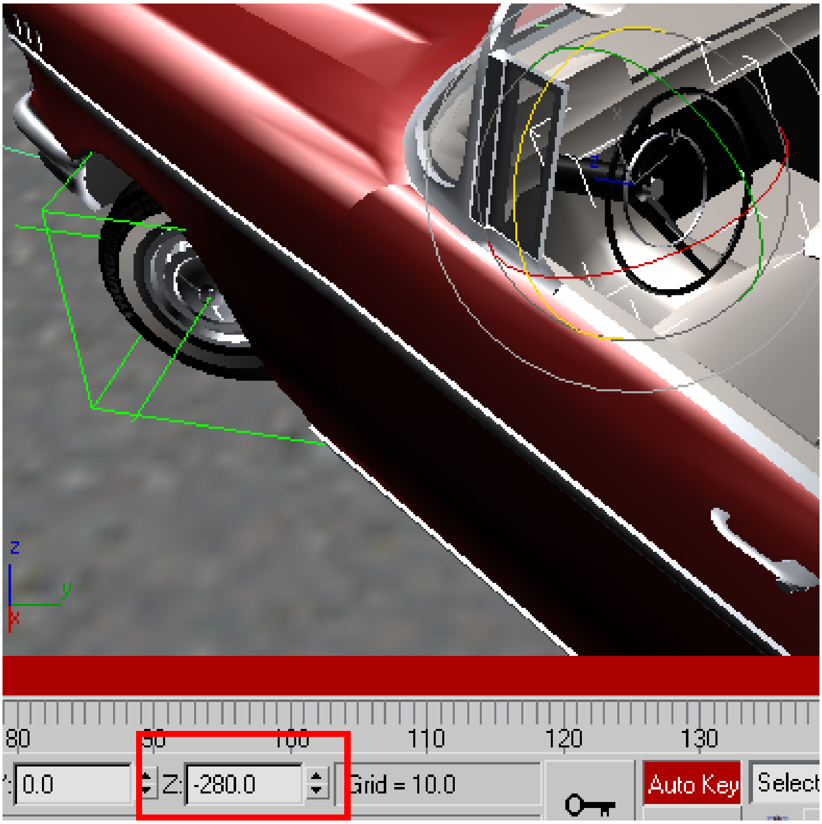
2  Switch to Top view, press F3 to switch back to Smooth + Highlights mode, then use Zoom Extents so the entire animation path is in view.

3 Go to frame 50, the point where the car is in the middle of its first turn.

4  Select *SWheel* and on the main toolbar and click Rotate.

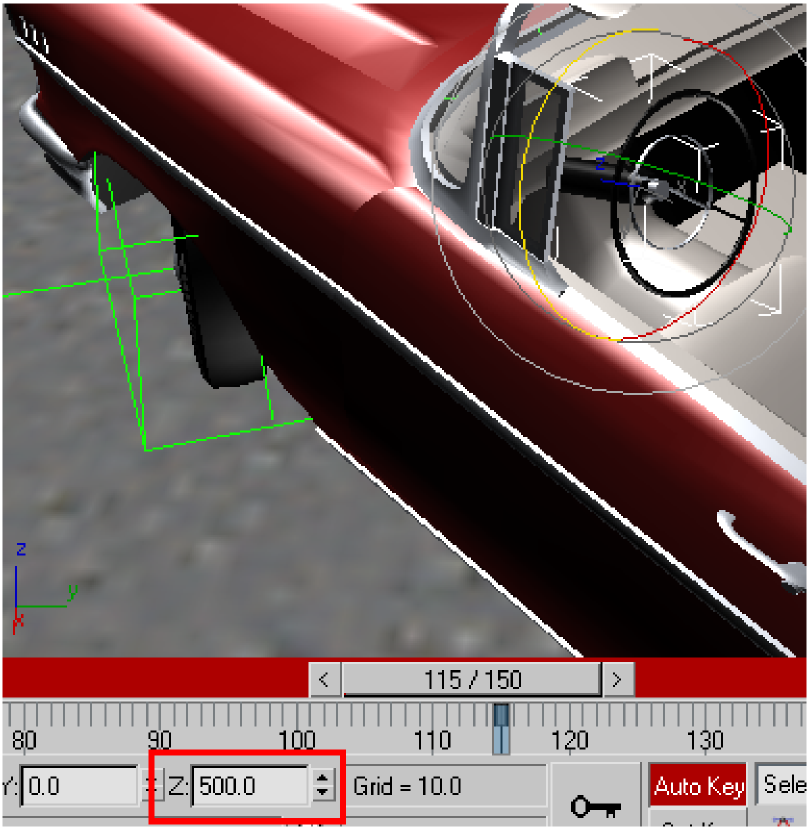
5 

Turn Auto Key on and rotate the steering wheel until the Z axis status bar reads -280.



Steering wheel rotation at -280 degrees about the Z axis

6 Go to frame 115, the point where the car is in the middle of the second turn, and rotate the steering wheel until the Z axis status bar reads 500.



Steering wheel rotation at 500 degrees in the Z axis

7 Go to the end of the animation and rotate the steering wheel until the Z axis status bar reads -220.

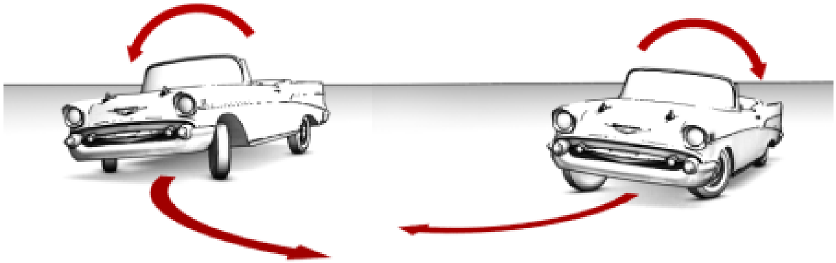
8 

Turn off Auto Key and test your animation.

9 Save your file as **mycar_rig_05.max**.

Setting Car Body Roll

Body roll is a phenomenon that occurs when a car rounds a sharp corner. This behavior is not usually apparent in modern day cars unless they are travelling at high speed. In older cars, such as our 1957 Chevy, however, the amount of body roll is discernible to both passengers and bystanders even when the vehicle is travelling at low speed.

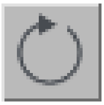


Set up for the lesson:

- Continue from the previous lesson or open *car_rig_05.max*.

Create the body roll effect:

In this lesson, you will create the effect of body roll by rotating the car along its local X axis. Roll direction will be based on the rotation of the steering wheel.

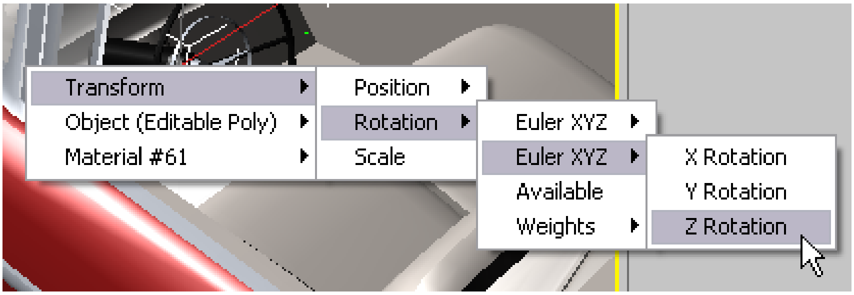
1. Make sure you are at frame 1 in your animation.
2. In the Top viewport, zoom in on the car and press F3 to turn Wireframe mode on.
3. On the main toolbar, click Rotate and make sure the coordinate system is set to Local.
4. In any viewport, select the steering wheel object.

   The steering wheel rotates about its local Z axis.
5. Select the car body object.

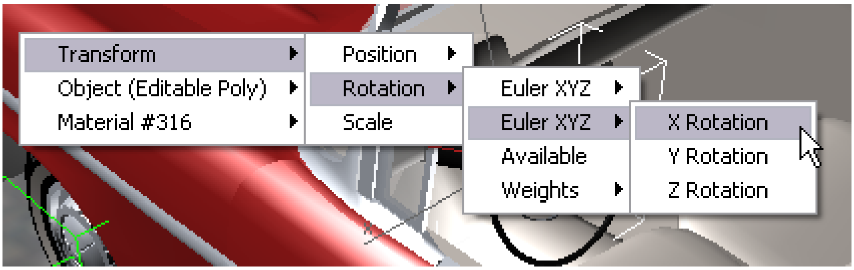
   The car body rolls about its local X axis.

6 Select the steering wheel again, then right-click on it and from the Quad menu, choose Wire Parameters.

7 Choose Transform > Rotation > (2nd) Euler XYZ > Z Rotation.



8 Select the car body and choose Transform > Rotation > (2nd) Euler XYZ > X Rotation.

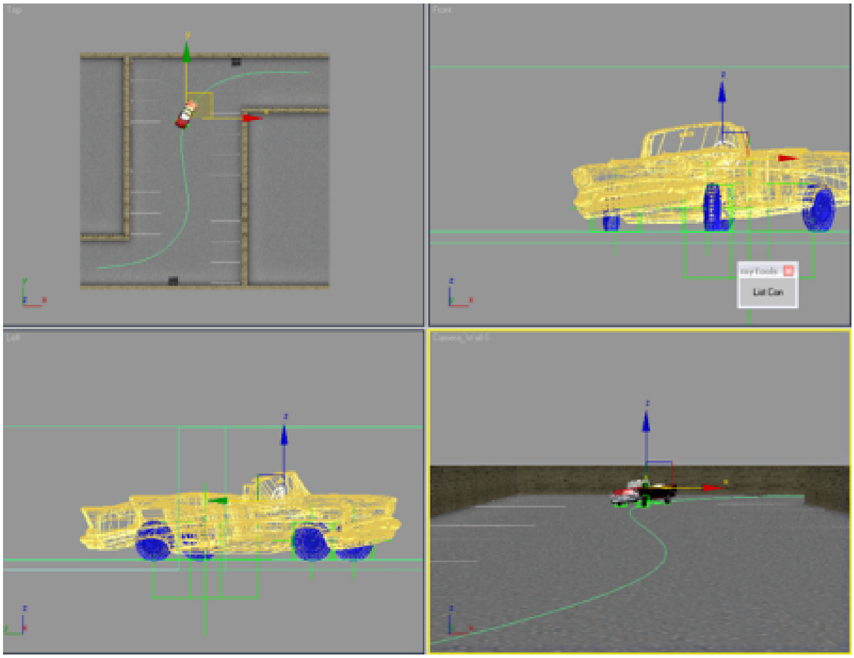


9 On the Parameter Wiring dialog, set the control direction to the right so the steering wheel rotation in Z controls the body roll in X.

10 On the right-hand Expressions panel, complete the expression so it reads: **Z_Rotation/40**, then click Connect.

NOTE The /40 factor in the expression divides the steering wheel rotation by 40 to ensure body roll rotation is significantly smaller than the rotation of the steering wheel. If you like, try experimenting with other values.

11 Click the Camera viewport label and choose Camera > Camera_Wall_S then scrub the animation to see the effect of the body roll.

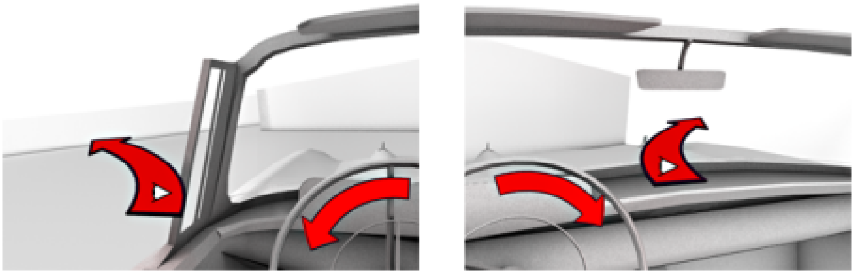


Body roll visible in Front, Left and Camera viewports.

12 Save your file as **mycar_rig_06.max**.

Adjust Driver Viewpoint

As a driver, when you use a steering wheel to initiate a turn, your eyes tend to follow the direction of the turn. When you turn left, you look left: when you turn right, you look right. In this lesson, your final task is to make the viewpoint of the driver react to the rotation of the steering wheel.

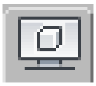


Set up the lesson:

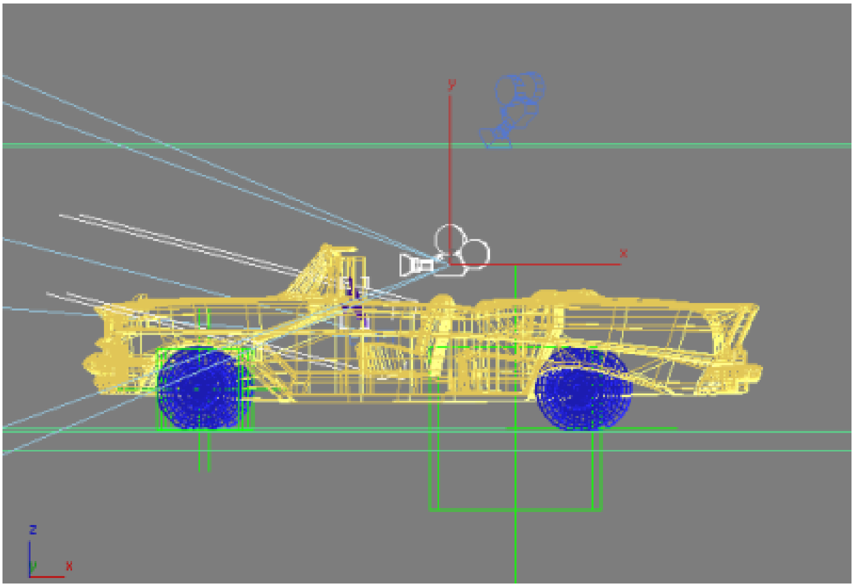
- Continue from the previous lesson or open *car_rig_06.max*.

Change the driver's point of view:

In this procedure, you will wire the rotation of the "driver view" camera to the steering wheel.

1  On the Display panel > Hide By Category rollout, turn off Cameras to re-display the cameras in the scene.

2 In the Front viewport, select the *Camera_Driver* object.



Camera_Driver object

This is the camera that occupies the driver's seat.

3 Click the List Con button on the myTools toolbar you created earlier in the tutorial.

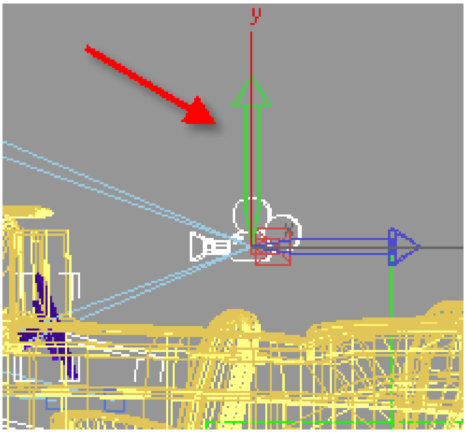
The List Con script automatically adds position and rotation list controllers, permitting you to retain control over the camera's local orientation.

NOTE If the myTools toolbar is not currently displayed on your interface, right-click a gray area on the main toolbar and choose myTools from the menu.

---

4 With the camera object selected, go to the hierarchy panel and on the Adjust Pivot rollout, click Affect Pivot Only.

Note that the swivel axis needed for the camera is the Y axis (displayed in green).



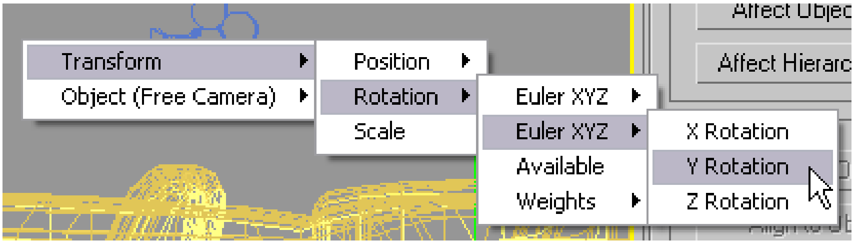
Camera swivel on Y axis

5 Click Affect Pivot Only again to exit pivot mode.

6 Select the steering wheel, right-click it, and from the menu choose Wire Parameters.

7 Choose Transform > Rotation > (2nd) Euler XYZ > Z Rotation.

8 Select the *Camera_Driver* object and choose Transform > Rotation > (2nd) Euler XYZ > Y Rotation.



9 On the Wiring Parameter dialog, set the control direction from left to right so the steering wheel controls the camera rotation.

10 On the right-hand Expressions panel, complete the expression so it reads: **Z_Rotation/10** and click Connect.

Keep the dialog open for now.

NOTE The /10 factor in the expression prevents the camera from rotating too far in either direction. You can experiment with different values to produce the results you need.

11 In the Camera viewport, click the label and choose Views > Camera Driver.

12 Scrub the animation to observe the wiring effect.

As the steering wheel rotates, the camera viewpoint swivels in the wrong direction.

13 In the Wiring Parameters dialog, add a negative operator in front of the expression, so that it reads: **-Z_Rotation/10**, then click Update.

14 Close the Wiring Parameters dialog and scrub the animation again.

The rig is now complete. To view a version of the finished product, open *car_rig_final.max*.

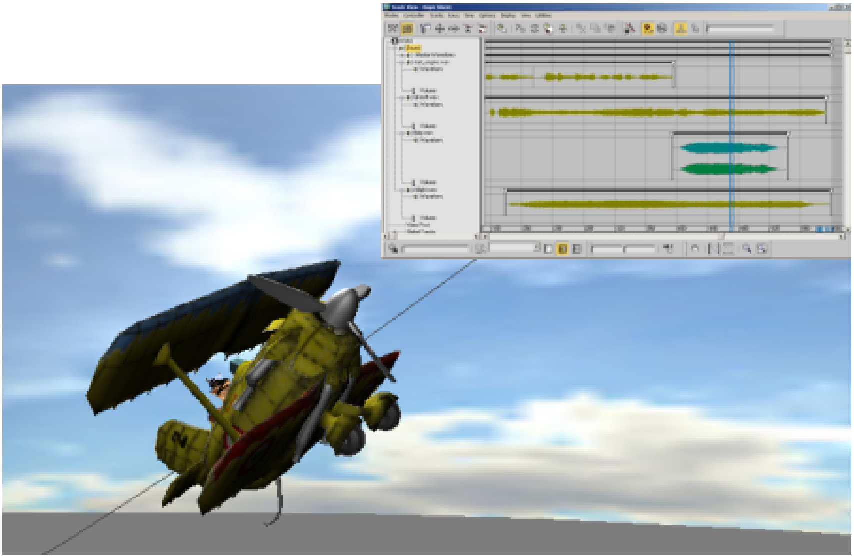
Summary

In this tutorial, you learned how to assign controllers to components of a model, and use expressions to ensure the controllers animate the components correctly. You also learned how to use Point helpers to animate a model along a path and were shown how to rework the model hierarchy so that a child object can respond to the animation of its parent.

Adding Sound Effects to Animation

In this tutorial, you have the scene of a World War One airfield somewhere in the north of France. A biplane is poised for takeoff, but it has no sound. Your task is to take a group of four *.wav* files and assemble them so that they play back as the plane rolls down the runway and takes to the sky.

You will set up the audio files in your scene, then adjust the sync in the Dope Sheet Editor.



In this tutorial, you will learn how to:

- Add audio files to a scene and control audio playback
- Use the Dope Sheet track editor to synchronize playback of audio clips with scene animation
- Use track controls to lengthen or shorten audio segments

Skill level: Beginner

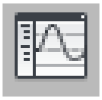
Time to complete: 20 minutes

Adding Sound Effects

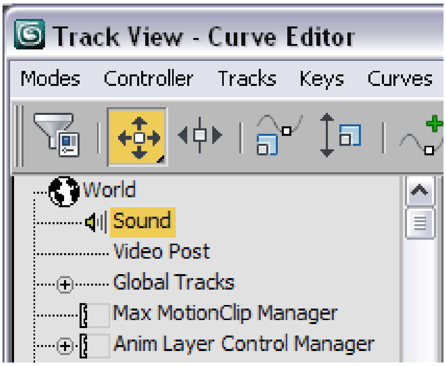
You will start by adding a number of audio files to your scene. Next, you'll display the files as clips on the Dope Sheet track editor and adjust their sequencing to fine-tune the audio playback.

Add sound files to your scene:

1  On the Quick Access toolbar, click the Open File button, navigate to the *animation\prosound* folder and open *prosound_start.max*, and open it.

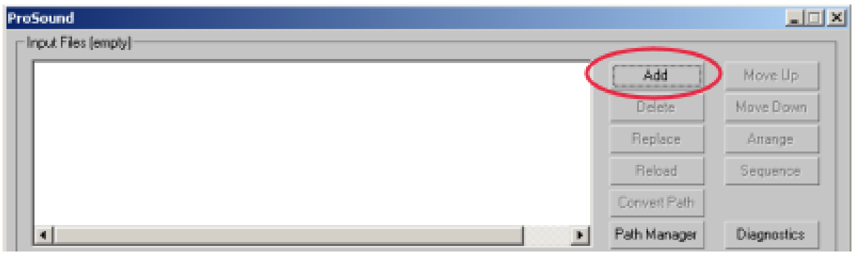
2  On the main toolbar, click Curve Editor (Open).

3 On the Curve Editor track view hierarchy, click the Sound track to highlight it.



4 Right-click and from the quad menu, choose Properties.

5 On the ProSound dialog > Input Files group, click Add.

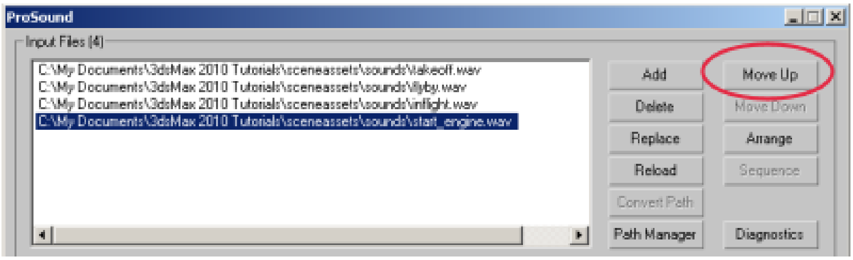


The Open dialog automatically opens to the sounds folder in your working directory. If the sound files *flyby.wav*, *inflight.wav*, *start_engine.wav*, and *takeoff.wav* are not there, navigate to the folder where you saved your ProSound tutorial scene files.

6 Highlight the *flyby.wav*, *inflight.wav*, *start_engine.wav*, and *takeoff.wav* sound files, in any order, then click Open.

The files display as a list in the Input Files group. While it is not mandatory, it can be helpful to place the files in the order in which their sounds are heard in the sound track.

7 From the Input Files group sound files list, highlight *start_engine.wav* and click Move Up as many times as needed to place the file at the top of the list.



8 Highlight *takeoff.wav* and click Move Up until it displays just below *start_engine.wav*. Use Move Up to place the *flyby.wav* next in the list.

9 Highlight a file in the list and in the File Details group, click the Play button to hear how it sounds.

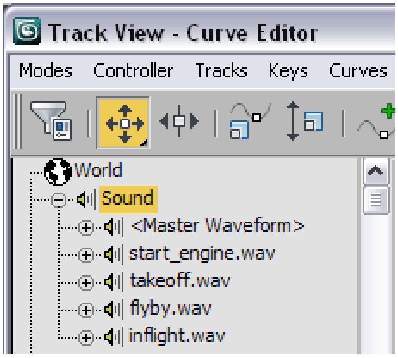
The File Details group also contains options that let you control how the file plays back in the scene. You can, for example, loop the file so it repeats a specified number of times, or enter the first and last frame on which the sound is played. The Active option lets you include or exclude the sound file in the playback.

In this tutorial, however, you will set these and other file playback parameters in a more visual way using the Dope Sheet.

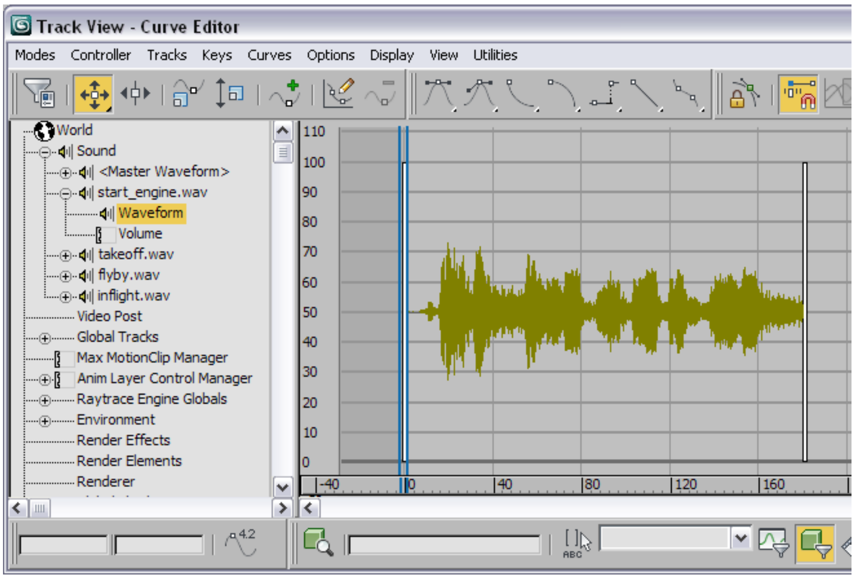
10 Close the ProSound dialog.

View the sound tracks in the Curve Editor and Dope Sheet:

1 On the Curve Editor track view hierarchy, expand the Sound track to display the master track as well as four additional tracks, each of which represent the sound files you loaded into the scene.



2 Expand each track. Each time you do so, highlight the track's Waveform component to display it.

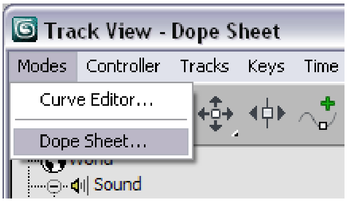


A waveform is an image that represents an audio signal, showing a change in amplitude over time. Soft sounds, like footsteps, produce a narrow pattern, whereas sharp sounds, like the scrape of a chair leg, show a wider pattern. Waveforms help you visualize the events in an audio recording.

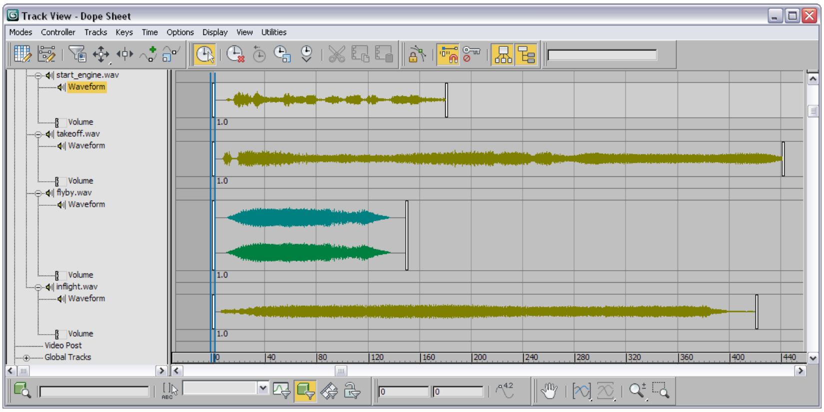
Notice how only one track waveform is visible at a time in the Curve Editor. (The master track waveform provides a visual compilation of all tracks.)

In this tutorial, it would be helpful to view all the waveforms together, so you can better adjust the timing of their audio segments. You can view multiple waveforms from the Dope Sheet.

3 On the Curve Editor menu, choose Modes > Dope Sheet.

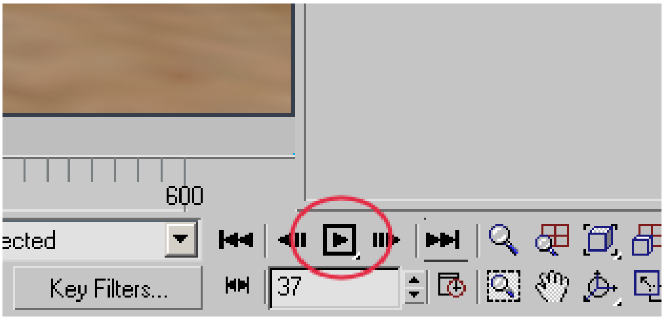


4 Collapse the master track to view just the waveforms and volume components of the four individual sound tracks.



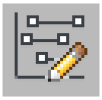
The first, second, and fourth sound tracks were recorded in mono and show a single waveform. The third sound track, *flyby.wav*, was recorded in stereo and displays two waveforms, one for its left and another for its right channel.

5 Play the animation forward to hear the sound files.

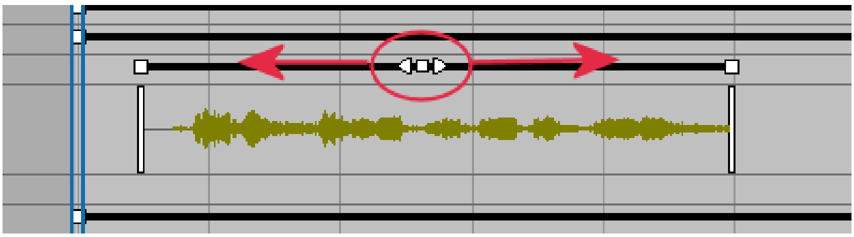


The segments overlap and produce a jumbled sound. You need to adjust the timing so each plays back at a more logical part in the animation.

Synchronize the audio with the animation:

1  On the Dope Sheet toolbar, click Edit Ranges.

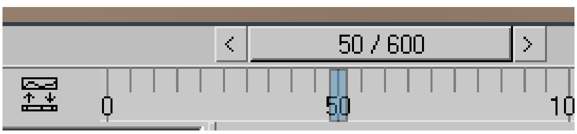
2 Click the *start_engine.wav* range bar and move it left and right to see how you can reposition the audio segment anywhere on the timeline.



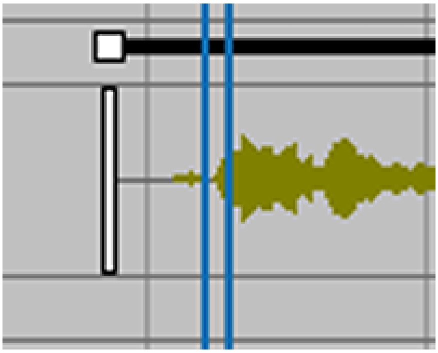
You need to reposition the start_engine segment to coincide with the part in the scene animation where the biplane propeller starts to turn.

3 Move the Dope Sheet window until the Camera01 viewport is visible.

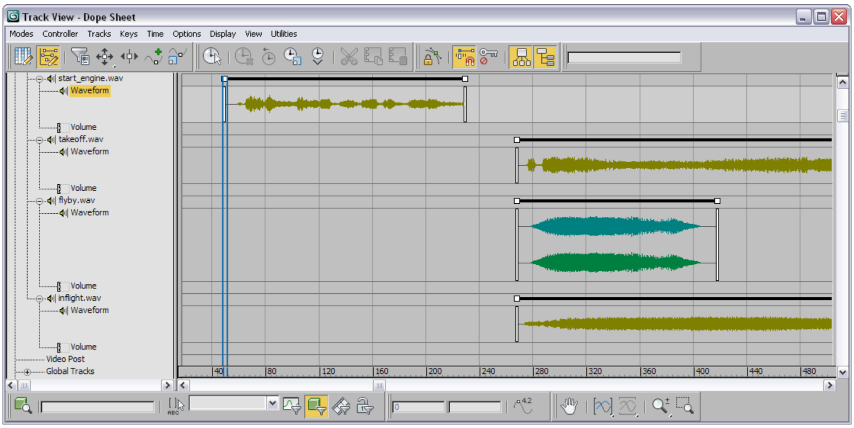
4 Drag the time slider until you see the propeller start to pick up speed, which is around frame 50.



5 Click the *start_engine.wav* range bar and drag it until the segment waveform is positioned at the vertical blue lines, which indicate the current frame.



6 For now, concentrate on just the *start_engine.wav* track by selecting the other track range bars in the Dope Sheet and moving them to the right, past frame 200.

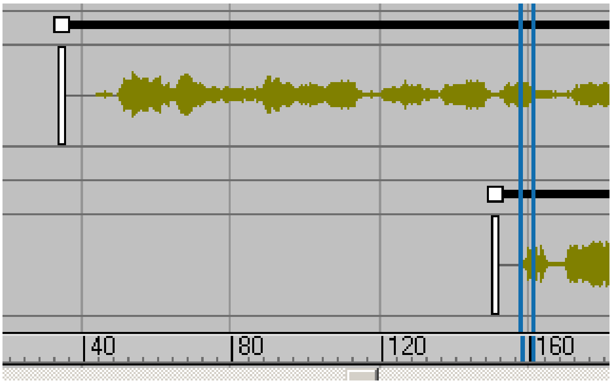


TIP Alternatively, you could temporarily mute the sound tracks in Track View by selecting their file names in the ProSound dialog and turning off the Active checkbox, as described in the previous procedure.

Scrub the animation.

The plane starts to move down the runway at frame 160. This is where you want to place the start of the *takeoff.wav* audio segment.

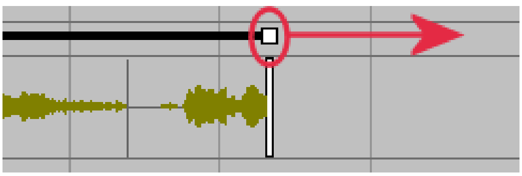
7 Click the *takeoff.wav* range bar and drag it until the segment waveform starts at frame 160.



As the plane begins to taxi down the runway it would be a good idea to prolong the start_engine audio segment, to simulate a sputtering takeoff.

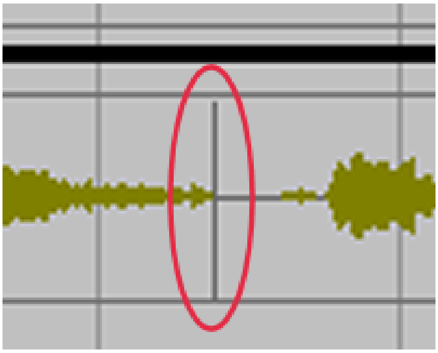
You could prolong the segment by going back to the ProSound dialog and in the File Details group, set Loops to repeat the segment as many times as needed. The next step shows you a different method.

8 On the Dope Sheet, click the right end point of the *start_engine.wav* range bar and drag to the right.

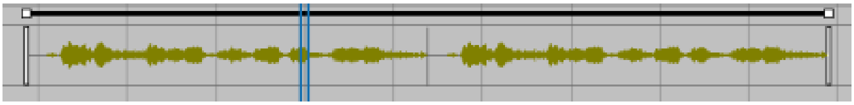


As you drag, the audio segment is repeated. You can drag for as many repetitions, or *loops*, as you like.

The end of each repetition is indicated by a vertical bar, as shown in the next illustration.



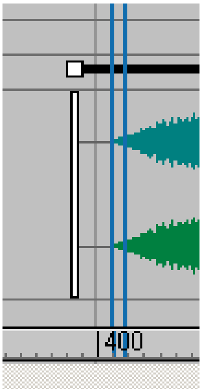
9 Continue dragging until the segment is repeated just once.



10 Move the Dope Sheet window until the Camera04 viewport is visible.

11 Scrub the animation again until, at around frame 405, you see the wheels of the biplane detach from the ground.

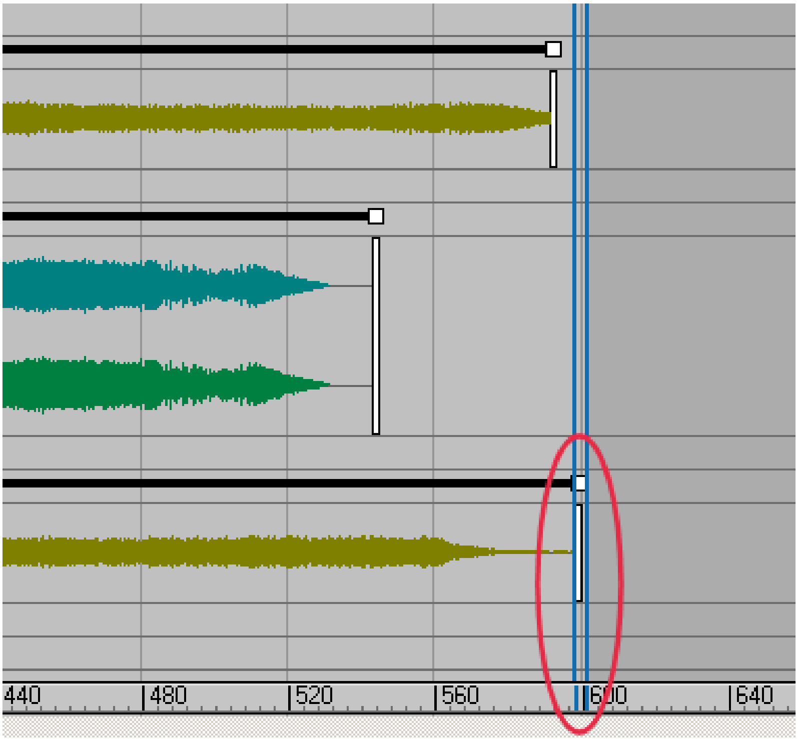
12 On the Dope Sheet editor, drag the *flyby.wav* range bar until the segment waveforms start at frame 405.



13 Scrub the animation again.

The waveform should peak at or around frame 435, when the biplane passes directly overhead.

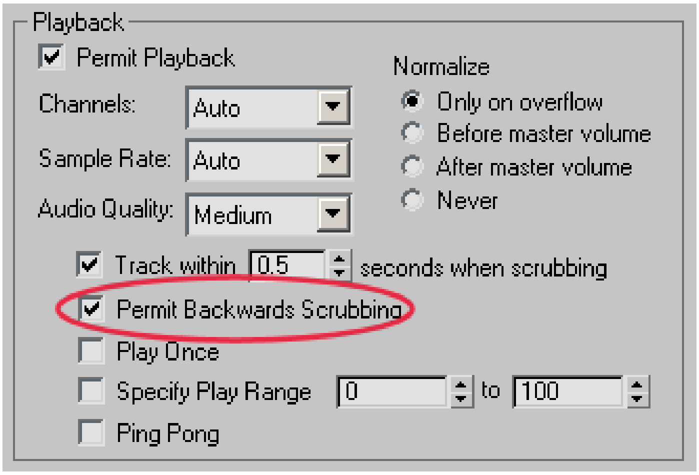
14 On the Dope Sheet, reposition the *inflight.wav* range bar so that the waveform fades out at the end of the animation, as shown in the next illustration.



15 Replay the animation to hear how all the audio segments fit together.

By default, the audio plays forward as you play the animation forward; there is no audio when you scrub animation in reverse.

16 To hear the audio in reverse when scrubbing, highlight, then right-click the Sound track in the Curve Editor or Dope Sheet and from the quad menu choose Properties. In the ProSound dialog > Playback controls, turn on Permit Backwards Scrubbing.



17 Try scrubbing the animation back and forth.

You can now hear all audio in the scene play in reverse.

18 Compare your work with the finished version of the scene by opening *prosound_completed.max*.

Summary

In this tutorial, you learned how to add multiple audio files to your scene and mix them in the Dope Sheet editor for playback during an animation.