

User's Guide



PhysX Plug-In for 3ds Max

September 2008

Table of Contents

Introduction	5
Installation	5
Getting Started	5
An Important Note about Scaling	5
Supported PhysX Features	6
The PhysX Control Panel	6
Initial Conditions	6
Rigid Bodies	7
Rigid Body Types	7
Specifying a Physical Geometric Representation	7
Grouping Physical Objects	8
Physical Parameters	8
Rigid Body Tutorial	
Triangle Meshes	9
Convex Hulls	10
Constraints (D6 Joints)	10
Steps to Create a D6 Joint	10
Creating Joints Using 'Joint Specification'	12
Deriving Joints from Animation Data	
Editing Joint Parameters Directly	13
Cloth	13
Making a Sphere Tear a Piece of Cloth	
Simulating Soft Objects with Cloth	
Metal Cloth	16
Fluids and Emitters	17
Fluid Drains	

	Connecting a Fluid to a Rigid Body	19
	Fluid Compartments	19
So	ft Bodies	19
Fo	rce Fields	20
VR	RD Connection	20
Expo	orting	21
Sam	nples	23
	Ballbounce	23
	Frictionramp	23
	clothBall	23
	Lincolnlogs	23
	mousehead	23
	simpleJoint	23
	SeeSawFun	23
	Catapultpivotanim	23
	Antigravity	23
	MetalBarrel	23
Que	stions and Answers	25
	Why don't I see my node hierarchy in the exported file?	25
	There are there multiple ways to define physics properties, which method works?	25
	What if someone else, without the PhysX plug-ins and DLLs, edits and saves my 3ds Max file? Will the physics specifications be lost?	25
	What about other modeling packages such as Maya or Softimage XSI?	25
	When we move a file from one modeling package to another we sometimes sacrifice information, usually the higher level information about our models. What will happen to the physics data?) 25
	Does our rendering tool chain have to migrate over to COLLADA in order to use the Physi exporter?	cs 25
	Why didn't you create subclasses instead of using user props?	26
	Why is the user property "Ellasticity" misspelled?	26
	I changed the user property "mass" but my change didn't show up. What happened?	26
	I like to use Max's groups for collections, other than compound rigid bodies, to help clear up and manage my scene. Can I do that without turning all my groups into single rigid	า
	bodies?	26

Why can't I change the mass of a group (compound rigid body)?	26
How can I make the name of a rigid body different from the shape inside used to describ	be
it?	26
What is the 'SW' button (inside the 'Geometry Tool' panel) for?	26

The NVIDIA PhysX Plug-in for Autodesk 3ds Max helps developers create and modify PhysX scenes directly in 3ds Max, without any additional programming. Scenes can then be exported for use in other applications, such as games.

Installation

The plug-in is available on the NVIDIA Developer Web site. Simply download and run the installer, which will automatically install the plug-in into the 3ds max folder. Please note that the plug-in requires that the PhysX SDK and appropriate PhysX driver are installed.

If you have a very old version previously installed, you might still have a PhysXLoader.dll or other PhysX-related dll's such as cooking in the 3dsmax folder. Please remove those manually.

Getting Started

You'll find the PhysX plug-in in 3ds Max's "Utilities" toolbar (pictured at right). It consists of three main components:

- PhysX Control Panel. Provides access to all major physics functionality. You'll be spending most of your time here.
- Geometry Tools. Offers settings for various subdivision levels and for creating convex hulls.
- **PhysX Parameters.** Covers general physics parameters that are common across physics simulations (for example, gravity). In general, you won't need to access this panel.

If you don't see these components, the plug-in was probably not installed correctly.

An Important Note about Scaling

Please note that the PhysX SDK does not currently support scaling of existing objects. So, for example, if you want to change the size of a sphere, then you must to do so by changing the radius of the sphere,

eventually see strange physical behavior after a few seconds of simulation.



Introduction

Supported PhysX Features

This chapter introduces the various features of the PhysX plug-In for 3ds Max. If you want to learn more about what the PhysX SDK is doing behind the scenes, we encourage you to read the PhysX SDK documentation.

The PhysX Control Panel

The PhysX Control Panel (pictured at right) is where you'll be doing most of your physics work. It consists of several buttons:

- Adding/Removing Objects from the Simulation
 - Add All Phys. Adds all physical objects to the simulation.
 - Add Selected. Adds the selected physical object(s) to the simulation.
 - **Remove All.** Removed all objects from the simulation.
 - Remove Sel. Removes the selected physical object(s) from the simulation.
- Controlling Simulation Playback
 - **Move Animation Slider.** If checked, 3ds Max's slider will move as simulation frames advance.
 - **Step.** Advances the physics simulation by a user-specified number of frames.
 - **Continuous.** Starts the physics simulation, or pauses the simulation if in progress.
 - **Reset.** Resets the physics simulation.
- Export. Allows you to export the current scene as XML, binary, or COLLADA.
- Make Physical/Edit. Creates a physical equivalent of a graphical object. When this button is pressed, the panel changes to show a number of additional physics options for the selected object. Note that you cannot add a graphical object to the physics simulation until you have pressed this button to create a "physical object" to represent the graphical object.

Initial Conditions

Please note that whenever you add physical objects to the simulation using the "Add All Phys." Or "Add Selected" buttons, the objects' positions are used as initial conditions.

– PhysX Control Panel	
Add/remove objects from sim:	
Add All Phys.	Remove All
Add Selected Remove Sel.	
Simulation:	
🔽 move animation slider	
Step	1 🔹 Frames
Continuous	Reset
Export:	
Export to XML 💌 Redo	
Simulate selected object(s) as:	
Make Physical / Edit	

Rigid Bodies



Figure 1 Rigid Body

The fundamental building block of a physics simulation is the rigid body (referred to as an "actor" by the PhysX SDK). This is a single physical unit that does not deform and has no internal moving parts. Every part of a rigid body moves in unison.

Rigid Body Types

You can define three types of rigid body:

- **Dynamic.** A dynamic rigid body interacts with other rigid bodies in the scene and is actively simulated with physics.
- **Static.** A static rigid body does not move, but does affect rigid bodies that come into contact with it. An example of a static rigid body is a ground plane for other objects to fall on or bounce off of.
- **Kinematic.** A kinematic rigid body obeys any animation that is applied to it, but is not actively simulated with physics. Dynamic physical objects do interact with it if they collide with it, but only the dynamic objects are affected by the collision.

Specifying a Physical Geometric Representation

In order for the PhysX SDK to simulate your scene, it needs to have a "physical" representation of the scene, as opposed to a "graphical" representation. The physical representation is what is actually used in the simulation, whereas the graphical representation is what someone would actually see.

For simple objects, such as a box, the physical representation can directly match the graphical representation. However, for more complex objects, a simpler physical representation may be used to maintain performance.

The geometry types that PhysX supports directly include spheres, capsules, boxes, convex meshes and general triangle meshes. Other types of geometry, such as cylinders, are approximated using convex or general triangle meshes.

Grouping Physical Objects

To get high performance in most physics engines, you should use convex pieces to define the collision volumes of rigid bodies. Therefore, it is often wise to break up a concave object into convex pieces and group them into a compound rigid body.

A compound rigid body is created by putting a number of objects together in a group. To do this, simply select all the relevant objects, go to 3ds Max's Group menu, and select "Group".

Grouping works within an articulated skeleton to simplify its physics model. For example, you may not want to physically simulate all the fingers of a hand. Instead, you would simply select the hand and its children (fingers) and group them all together.

Similar groupings can also work with the internals of a skeleton. There may be a series of vertebrae bones used for skinning and to animate subtle gestures. These could be grouped into a single rigid body for physical simulation.

Physical Parameters

Rigid bodies have a number of physical properties that can be specified, including mass, coefficients of friction, coefficient of restitution, and more. Details are provided in the descriptions of the utilities in later sections of this document.

Rigid Body Tutorial

The best way to become familiar with plug-ins is to actually use them. Please try the following steps to create a simple rigid body simulation.

- 1. Create a sphere.
- 2. Go to "Utilities" and then "PhysX Control Panel"
- 3. Select the sphere you previously created.
- 4. Press the "Make physical / edit" button. This will create a physical equivalent of your object.
- 5. A new set of controls will now appear. In the dropdown, you can specify the type of object that you are creating. This will default to "dynamic rigid body".
- 6. In the new panel, you can change properties such as mass and coefficients of friction.
- 7. When you are satisfied with the properties of the rigid body you are creating, press the "Add Selected" button. This adds the selected object to the physics simulation. There is also an "Add All Phys." button, which adds all physical objects to the simulation. "Remove All" and "Remove Selected" buttons allow you to remove physical objects from the simulation.
- Press the "Continuous" button in the PhysX panel. This will start the physics simulation.
 You can also use the "Step" button to advance the simulation one step at a time. The "Reset" button resets the simulation to its initial state.
- Now the sphere should start falling down, due to gravity, which by default is 9.81 in the -Z direction. (The gravity direction and amplitude can be set in the "PhysX Parameters" panel.)

- 10. Now let's create a floor for the sphere to fall on. First, reset the simulation state by pressing the "Reset" button.
- 11. Create a large, thin box (at the height of 3ds Max's default grid).
- 12. Select "Make Physical /Edit" for this object, but this time, make sure that it is a "Static Rigid Body".
- 13. Add it to the PhysX simulation by clicking "Add Selected".
- 14. Make sure the sphere is translated upwards so that it will fall on the floor. (Your scene should now look similar to the picture at right).
- 15. Press "Continuous" to play the simulation, and watch the sphere fall on the floor.



You've now created a simple physics simulation. Each rigid body in the scene corresponds to an instance of the PhysX SDK's NxActor class.

Triangle Meshes

You can use arbitrary triangle meshes for static physical objects. (Using triangle meshes for dynamic physical objects can be tricky because of the meshes' arbitrary shapes.)

Sharing Triangle Meshes

If you need to make multiple copies of the same triangle mesh, you should do that by using 3ds Max instances. Please note that scaling will not work here, so if you want to make differently-sized instances of the same triangle mesh, you will need to make several triangle meshes.

Follow these steps to share a triangle mesh:

- 1. Create a triangle mesh (for example, a torus, which is not natively supported in PhysX as a basic object type)
- 2. Make a 3ds Max instance by copying and pasting the object (choose "Instance" when prompted)
- 3. Press the "Make Physical / Edit" button to create a physical object.
- 4. Make sure it is a "Static Rigid Body"
- 5. Under "Specify Physics Representation", choose "Triangle Mesh"

Performance Advice

For large scenes it is important that you manage to share triangle meshes (if there are instances of the same mesh). Please note that now you should not use triangle meshes for dynamic objects.

Convex Hulls

For complex dynamic rigid bodies, you will want to use a simplified triangle mesh in order to get a convincing simulation while maintaining performance. The most common solution for this is called a convex hull, which has no concave sides.

To create a convex hull, you can either add a simple triangle mesh object and not specify that it should be simulated as a triangle mesh (the "important" point from above) or you can make sure that you know how the convex hull will look like, by creating an explicit convex hull explicitly, through a "Geometry Tool" available in the PhysX Utility.

- 1. Create a torus
- 2. Select the torus
- 3. Go to "PhysX Utility" panel and click the "Geometry Tools" button. A new panel will open below the button.
- 4. At the bottom of that panel there is a heading called "Create convex hulls". Here you can choose the number of vertices for the hull, as well as the amount of inflation. Increasing the inflation can help to prevent unwanted intersections between the original mesh and the convex hull.
- 5. Click on "Create Hulls" to create your new convex hull. This will be a new object in the physical simulation.

Constraints (D6 Joints)

Constraints allow you to join rigid bodies together while controlling their degrees of freedom. PhysX supports the "6 degrees of freedom joint", which is a versatile single joint object that can be configured to work as most other joint types.

Steps to Create a D6 Joint

- 1. Create an object (such as a sphere), and make it a Dynamic Rigid Body
- 2. Create a second object (such as a box), and make it a Dynamic Rigid Body

We'll be connecting these two objects with a constraint.

3. Choose 'Helper' in 'Geometry' tool as the following picture

	: L. 🛞 🖾 🏌
0 7) 🌾 🛱 🔽 🚿 🐐
NVIDIA	A PhysX 🗾
	Ohiosh Turas
- n	
	AutoGrid
rL-	AutoGrid

Figure 2 Helpers

- 4. Choose 'NVIDIA PhysX' inside the drop-down list. It will show a sub-panel featuring 3 buttons, 'PxFluid', 'PxEmitter' and 'PxJoint'.
- 5. Press button 'PxJoint' and create the joint object. Note that when creating the object, you should click and drag to define the joint's degrees of freedom.
- 6. Choose the Joint that you created and choose the 'Modify' Tool. You'll then see the following panel. 'Body0' and 'Body1' specify the two objects that this joint constrains.

-	Constraint
Body0:	undefined <u>x</u>
Body1:	undefined <u>x</u>
🗖 Brea	kable
Max For	ce 10.0 🛊
Max To	que 10.0 韋
🗖 🗖 Colli	sion Enabled
Projectio	on none 🗾
Projectio	on Dist 🛛 🚺 韋
Projectio	on Angle 🛛 0.087 🛛 韋
🗖 🗖 Gea	ring
Gear Ra	itio 1.0 ≑
Attach	ment point:
Joint	center 🗾 pick

Figure 3. Modifying the Constraint

- 7. Press button 'undefined' beside label 'Body0' and then choose the sphere
- 8. Press button 'undefined' beside label 'Body1' and then choose the box
- 9. Modify other parameters of the Joint like swing, twist etc., to your liking.
- 10. Add all of them to physics

Please see the 'simpleJoint' sample project that accompanies the plug-in for an example.

Creating Joints Using 'Joint Specification'

The Joint Specification rollout specifies the joint constraint between the current object and the current object's parent. Should the object not have a parent, then the constraint is with respect to the static world. Note that not all joint-body systems can be built this way but it is a good way to build many systems such as biped rag dolls or single objects with simple constraint behavior such as a door on a hinge.

Joint Specification is not shown directly after running Max. First click the MAXScript button in Utility page (Hammer). Then in the MAXScript rollout, choose 'Joint Specification' in the combo box, as shown below:

N 🖉 🔠 🛞 🖉	
- Utilities ;	- MAXScript
More Sets	Open Listener
Asset Browser	New Script
Camera Match	Open Script
Collapse	
Color Clipboard	Run Script
Measure	Utilities
Motion Capture	Convert to mr Area Lit
Reset XForm	Convert to mr Area Lights Joint Specification
MAXScript	PhysX Utility
reactor	

Figure 4 Run 'Joint Specification'

Deriving Joints from Animation Data

If the movement characteristics of an object are simple enough then the easiest way to specify this is for an artist to simply animate the object through its range of motion. In other words, the way something moves in Max is how it should move in the game.

Right now there are two options for calculating a joint from animation. If you know that you have a hinge joint (1 degree of freedom) then click on the "Calc Hinge Joint From Anim" button. If the joint requires more maneuverability then calculate a sphere joint from it. These routines cycle through the key frames and try to determine the joint characteristics. Single 1 degree of freedom joints (hinges) are easiest to derive, even for an arbitrary axis of rotation. So you don't even have to try to reorient the pivot to align it with the hinge. Just animate the object – that's all. More complicated rotational joints present more of a mathematical challenge due to the non-orthogonal nature of orientation space. Calculating a sphere joint is useful for computing the limits of motion for that type of joint.

Please note that currently, it is best to keep your pivots aligned with the node's coordinate frame. For any geometry that is part of a group (compound rigid body) please keep the pivot at the node's origin. There is no benefit to having an offset pivot to something that can't move.

To speed up artist productivity, the joint editor can be applied to multiple selections. For example you can select the forearms and calves and calculate a hinge joint from the animations, then select the upper-arms and thighs and calculate sphere joints.

Editing Joint Parameters Directly

Individual joint parameters can be modified using a collection of UI elements that map directly to the joint specification part of the PhysX SDK data model.

Cloth

Cloth can be created through any polygonal object, such as a plane (for a piece of fabric) or a sphere (for a soft ball).

Making a Sphere Tear a Piece of Cloth

- 1. Open 'Geometry' Tool and select 'Standard Primitives'
- 2. Create Sphere and a Plane. We'll make the plane into cloth, and have the sphere tear through it.
- 3. Move the sphere so that it is directly above the plane, ready to fall through it.
- 4. Select the sphere and convert it into a Dynamic Rigid Body.
- 5. Select the plane. Make sure that it's made of about 15 segments in the length and width dimensions. Having more segments makes the simulation more realistic.
- 6. Now convert the plane into an editable mesh, as shown below (right-click on it, select Convert To -> Convert to Editable Mesh).



Figure 5 Convert to editable mesh

7. Now you will see the mesh editor, shown in the following image:



- 8. In this case, you don't need to tessellate the plane further. But if you had to, you could do the following:
 - \circ $\;$ Choose the Solid Red Triangle which stands for 'Face'
 - o Click button 'Tessellate' which is under 'Edit Geometry' category
- 9. Now go back to the PhysX panel in the 'Utilities' Tool. Press the 'Make Physical/Edit' button in 'PhysX Control Panel', and choose "Cloth".

_[- Phys≯	(Utility
PhysX SDK Version 2.8.0	
PhysX Co	ntrol Panel
Geometry Tools	
PhysX Parameters	
Close	
– PhysX Co	ntrol Panel
Add/remove ob	jects from sim:
Add All Phys.	Remove All
Add Selected Remove Sel.	
Simulate:	
Step Contin	uous Reset
Export:	
Export to XML - Redo	
Simulate selected object(s) as:	
Make Physical / Edit	

10. Now check the "Tearable" checkbox in the list of cloth properties. This will allow the sphere to tear the cloth as it falls through.

- Cloth Properties
Cloth Flags:
🗖 Pressure 🔲 Static
Disable col. Self collision
🔲 Gravity 🔲 Bending
🔲 Ortho bend 🔲 Damping
🔲 2-way coll. 🔲 Triangle coll.
🔲 Tearable 🔲 Hardware
Cloth Properties:
Auto-attach to shapes
Thickness 0.01 💲
Density 1.0 🜻
Bending Stiff. 1.0 😫
Stretching Stiff. 1.0 😫
Damping 0.5 😫
Friction 0.5
Pressure 1.0
Tear factor 1.5
Collision Resp 1.0
Attachment Resp 1.0
Solver Iterations 5

11. Click "Add All Phys." to add both objects to the simulation, and then click "Continuous" to play the simulation. You should see the sphere tear through the cloth.

Now, the cloth is created and you can try to simulate and watch it to change in shapes.

This example is also available as the "clothBall" sample project.

Simulating Soft Objects with Cloth

You can actually use cloth to simulate a soft enclosed object (such as a beach ball). To do this, simply convert the object into a cloth (but make sure that the object's volume really is closed). Then enable "Pressure" in the list of Cloth Properties, and adjust the "Pressure" value in the Cloth Properties.

For more complex soft objects, you should use a "soft body", which is described later in this guide.

Metal Cloth

Metal Cloth is a special case of cloth that is useful for simulating deformable metal items like barrels.

Metal cloth is very easy to create:

- 1. Create an object (such as a box)
- 2. Click button 'Make Physical/Edit' button in the PhysX control panel
- 3. Choose 'Metal Cloth' as the simulation type
- 4. Modify the Metal Cloth properties
- 5. Add it to the physics simulation
- 6. Start the simulation

There is a sample project called 'metalBarrel' that demonstrates metal cloth in action.

Fluids and Emitters

Fluids are created a little differently from the other objects we've covered so far. We'll be using the "helpers" in 3ds Max. These can be found in the "Create" part of the right-hand toolbar. Then, choose "NVIDIA PhysX" from the list of helpers.

Fluids are created with two types of object: a PxFluid object (which defines the fluid type and properties) and a PxEmitter (which defines where the fluid particles appear from). You can associate multiple PxEmitters with a single PxFuild object. That way, you can change the PxFluid once to affect all the PxEmitters associated with it.

The following steps describe how to create a simple fluid that will spray some fluid particles into the air. If you want to learn more about how to set the properties of the fluid, please read the PhysX SDK user documentation.

Let's create a simple fluid that has one emitter:

- 1. Go to 3ds Max's "Create" toolbar
- 2. Select "Helpers" and then "NVIDIA PhysX"
- 3. Create a "PxFluid" object. The positioning of this object is not very important but it's generally convenient to place it close to where you'll put the emitter, since they work together.
- 4. Create a "PxEmitter". This defines where your fluid particles will start from.
- 5. Next, you need to bind the emitter to the fluid. Select the emitter and then go to the Fluid parameters panel.
- 6. Click the button 'undefined' after label "Fluid:" (see following picture) and then select the fluid object.

- Fluid Params		
Fluid:	undefined	
Frame reference:	undefined	
Emitter Type		
Constant Flow Rate 📃 💌		
Shape Rectangular 💌		
Dimension		
× 0.25 € Y 0.25 €		
 Enabled Force On Body Add Body Velocity Visualization 		
Max Particles: 👖	000 💲	
Fluid Velocity	.0 🛟	
Rate 100.0 \$		
Particle Lifetime 0.0		
Repulsion Coefficient		
[1	.0 🛟	
Random Position	¢ z 0.0 ¢	
Random Angle 🕻).0	

Figure 7 PxEmitter Properties

- 7. Now go to the PhysX control panel and add these objects to the simulation by clicking "Add All Phys."
- 8. Start the simulation (by pressing the "Continuous" button). You should see a small chain of particles falling down from the emitter.

Now you can play around with the fluid and emitter settings. Most of the settings are direct copies of what can be done from within the PhysX SDK, so please refer to the SDK documentation if you are wondering how you should set the properties.

Fluid Drains

A fluid drain is an object that, when hit by a fluid particle, removes that particle from the physics simulation. To create a drain:

- 1. Create a rigid body, such as a box, which will be the fluid drain shape.
- 2. Create a fluid as above, with an emitter
- 3. Edit the fluid properties, at the bottom there is a list of "fluid drains"

- 4. Click "add drain" and select the box
- 5. Now you have a drain. You can add or remove drains as you like.

Drains can be shared amongst fluids. You do not have to create specific objects to work as drains (e.g., had a floor already, you could use that as drain).

Connecting a Fluid to a Rigid Body

It is possible to create fluid emitters that are moved around together with an object. To do this:

- 1. Select the emitter
- 2. Edit the properties of the emitter
- 3. Click the button next to "Frame reference" and then select a rigid body

The fluid emitter will now follow the selected rigid body as it moves around in the world.

Fluid Compartments

There is a setting called "fluid compartment" in the fluid settings panel. This is a simple way of creating fluids in different compartments (basically different scenes). 0 is the default compartment. If you want a fluid to be created in another compartment, select a different number and then add the fluid. If you want to change the compartment number of a simulated fluid, you should first "remove all", change the compartment number, and then "add all".

Soft Bodies

The softbody is a type of physics object that is built of tetrahedra and simulated in a way that makes the object seem "soft". Because it can be difficult to create soft bodies, the plug-in offers some tools to help with soft body authoring.

Follow these steps to create a soft body:

- 1. Create a geosphere (this is a more stable choice for softbody than a normal sphere)
- 2. Go to the PhysX Utility and open the "geometry tools" panel
- 3. Select the geosphere
- 4. Press the "build mesh" button (with "create editable mesh" and "create iso surface" selected, subdiv 10 and simplification 0,5)
- 5. Now you will get a new sphere, which is a bit larger than the old one
- 6. Select the new sphere
- 7. Go to the PhysX control panel
- 8. Make physical
- 9. Select "SoftBody"
- 10. Add all
- 11. Then simulate. Currently you can't see the progress of the simulation inside of 3ds Max; however, you can see it in VRD. Below you can see how to use VRD for watching the simulation.

Force Fields

Force fields allow you to impact forces on rigid bodies that enter their area of influence. Force fields are supported by the plug-in only for SDK 2.8.0 and later versions.

Please follow the steps to create Force Field.

- 1. Create a box
- 2. Choose 'Utilities' Tool, which is shown as a 'Hammer'
- 3. Click button 'Make Physical/Edit' and choose 'ForceField'
- 4. Now a force field emanates from the box, and you can edit its properties.

You can run Sample 'antigravity' to see a rigid body ball going up and down inside and outside of a box-shaped force field.

Please note that force field parameters are complex. Therefore, we highly recommend that you read the PhysX SDK documentation to fully understand how force fields work.

VRD Connection

The 3ds Max plug-in supports external analysis by the NVIDIA PhysX VRD (Visual Remote Debugger). The VRD is a useful tool that allows you to understand exactly how the PhysX SDK sees your scene from a physics point of view.

Follow these steps to examine your simulation with the VRD:

- 1. Start the VRD
- 2. Remove all physics objects from the simulation (this will also release the SDK) via the "Remove All" Button in the PhysX Control Panel.
- 3. Re-add all physics objects to the simulation via the "Add All Phys." button in the PhysX Control Panel.
- 4. Start the simulation. You should now be able to follow along in the VRD

Please note: You cannot currently drag objects around inside the VRD; this will crash the plug-in. A future release will fix this problem.

The plug-in supports the exporting of PhysX Objects in three formats: COLLADA, XML, and Binary. The exporting function is based on NxuStream.

The "Export" function is available through the 'PhysX Control Panel'. Simply choose the file type first and press the 'Export to' button to perform the export.

Exporting

Ballbounce

This project shows three balls with different properties, which cause them to have different behaviors after falling. The higher a ball's Restitution is, the higher it bounces.

Frictionramp

This project shows rigid bodies with different friction values.

clothBall

This project shows a cloth sphere.

Lincolnlogs

This is a simple demonstration of compound objects created in 3DS Max. Each log is a compound object made out of boxes which can interlock for a secure fit.

mousehead

The Mouse head sample consists of a single rigid body. This body consists of five spheres, but only four of the spheres are visible. The fifth sphere is small and is embedded inside the large sphere near the bottom of the Mouse head. This hidden sphere has a mass many times that of the others, which dramatically lowers the center of gravity. This is what causes the mouse head to settle at rest in an upright position.

simpleJoint

This project shows a simple constraint (D6 joint).

SeeSawFun

This project shows a more complex joint, created using Joint Specification.

Catapultpivotanim

This project shows another complex joint, created using Joint Specification.

Antigravity

This project shows a Rigid Body falling strangely as it passes a force field that is set with antigravity force.

MetalBarrel

This project shows a heavy ball falling on a metal barrel, causing the barrel to deform.

Samples

Questions and Answers

Why don't I see my node hierarchy in the exported file?

A collection of rigid bodies might be represented in a tree hierarchy in 3ds Max. For example, the (fattened) bones of a biped will be become the rigid bodies for a rag doll. In this 3ds Max context it makes most sense to think of positions and orientations of a node in the parent's frame (coordinate system). This parent-child relationship is not meaningful in a dynamics setting. The relationships between rigid bodies are nothing more than physical constraints such as joints and body-body collision rules. Therefore rigid bodies are exported in model or global coordinates and hierarchy information is not reflected in the physics model.

There are there multiple ways to define physics properties, which method works?

The best approach is to use the PhysX Properties Editor. You can also specify properties such as the mass of a shape using Reactor's dialog or the user property editor.

What if someone else, without the PhysX plug-ins and DLLs, edits and saves my 3ds Max file? Will the physics specifications be lost?

No worries. Edit away. That's why we used the user property facilities of max instead of relying on modifier variables.

What about other modeling packages such as Maya or Softimage XSI?

PhysX is built into version 5 of XSI Essentials and Advanced.

For Maya we also provide a plug-in on our developer web site (<u>http://developer.nvidia.com</u>).

When we move a file from one modeling package to another we sometimes sacrifice information, usually the higher level information about our models. What will happen to the physics data?

We are working with the COLLADA consortium to standardize the physics representation. The objective was that by using COLLADA, the physics information would not be lost when using different applications to edit the same models. However this is a long term goal.

Does our rendering tool chain have to migrate over to COLLADA in order to use the Physics exporter?

No, this isn't a requirement. Within a COLLADA file, the physics data sits separate from, but parallel to, the rendering data. The physics data will easily work with your existing graphics data in any format. There are also other export options including PML and ASCII/binary serializations as well as the ability to write custom exporters by traversing 3DS Max's nodes.

Why didn't you create subclasses instead of using user props?

In addition to the problems with competing versions and sharing .max files with others, using subclasses would limit the types of objects that can become physics objects. It is better to allow any type of geometry be used as a rigid body.

Why is the user property "Ellasticity" misspelled?

This is in order to interface seamlessly with Reactor. When data is exported, this is corrected to properly fit the physics data schema.

I changed the user property "mass" but my change didn't show up. What happened?

Be careful here. Max is usually case insensitive. However, the user property hash strings are case sensitive. Look for "Mass" instead of "mass".

I like to use Max's groups for collections, other than compound rigid bodies, to help clean up and manage my scene. Can I do that without turning all my groups into single rigid bodies?

Unfortunately, the plug-in does not support that right now.

Why does setting mass to 0 make something static?

Mathematically speaking the mass should be set to infinite. However, the user interface elements of max (such as spinners) don't support that. Since zero mass is meaningless, it is a convenient way to specify something is static. The best thing to do is to specify 'static', 'dynamic', or 'kinematic' using the radio selection on the PhysX properties editor.

Why can't I change the mass of a group (compound rigid body)?

The mass of a group is a derived value. Mass is specified in each of the shapes that make up the compound rigid body. A physics engine has to also know how the weight is distributed within an object in order to compute other important inertial properties that affect how an object rotates.

How can I make the name of a rigid body different from the shape inside used to describe it?

Simply Group the node. Yes, you can have a group of one. The name of the group is the name of the rigid body, while the name of the initial node determines the shape name.

What is the 'SW' button (inside the 'Geometry Tool' panel) for?

"SW" stands for "skin width". It uses the current SDK skin width to set the skin width. If current SDK skin width is not available, then it sets the value as 0.025.

Notice

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA, the NVIDIA logo, and FX Composer are trademarks or registered trademarks of NVIDIA Corporation in the United States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated

Copyright

© 2008 NVIDIA Corporation. All rights reserved.



NVIDIA Corporation 2701 San Tomas Expressway Santa Clara, CA 95050 www.nvidia.com