Vanda Engine 1.3.3

Next page →

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

Vanda Engine 1.3.3

User guide

Vanda Engine 1.3.3

 \leftarrow Previous page Next page \rightarrow

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

Vanda Engine 1.3.3

Reference Manual

The contents of Vanda Engine 1.3.3

Vanda

Made with help of Dr.Explain

Þ

Vanda Engine 1.3.3 \rightarrow Vanda	\leftarrow Previous page Next page \rightarrow
CODO: To change the footer's content go to Dr Evolain menu Ont	

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Vanda

,	Introduction
~	Learning the GUI
,	Modify Menu in Depth
,	Engine Objects
,	<u>Physics</u>
,	Tools
,	<u>Scripting</u>
,	Publishing
,	Exporting
	Reference
~	FAQ

<u>Vanda Engine 1.3.3</u> \rightarrow <u>Vanda</u> \rightarrow Introduction	← Previous page <u>Next page</u> →
	2
TODO: To change the footer's content go to Dr.Explain menu Options : Pr	oject Settings : HTML (CHM) Export : Setup

t go to Dr.Explain menu **Options : Project Settings : HTML (C** HM) Exp HTML Template and Layout

The contents of Introduction

- Licence Agreement
- An Overview to Vanda Engine

= \mathbf{F}



Licence Agreement

The MIT License (MIT)

Copyright (c) 2014 Ehsan Kamrani

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.



An Overview to Vanda Engine



In this document, we beriefly explain the basic concepts of Vanda Engine.

VScenes

In Vanda Engine, you should think of Vanda Engine Scenes (VScenes) as individual levels, or areas of design content. By constructing your design with many scenes, you'll be able to distribute loading times and test different parts of your design individually.

Any currently open VScene is what you are working on, as no two VScenes can be

worked on simultaneously.

A VScene is a collection of external scenes in COLLADA format, a PhysX XML scene or built in physics elements, built-in objects, rendering effects, extra textures and other elements that are critical for a real time interactive scene.

You can use scripting to load another VScene while you are in current VScene. Such scenes are called Chained VScenes.



VScene



This diagram depicts different elements of a VScene

External Scenes in COLLADA Format

Geometry

A geometry is a collection of vertices, edges, normals, triangles, tangents, bitangents, UVs, materials, physics, and several textures such as normal map, alpha map and shadow map.

Vanda Engine reads vertices, edges, triangles, tangents, bitangents, multiple UVs, and materials of the geometries from 3D tools capable of exporting to COLLADA. However, Vanda Engine interprets each geometry as an Instanced geometry which is a more general concept in computer graphics.

Instanced Geometry

geometry instancing is the practice of rendering multiple instances of the same geometry in a scene at once. Although geometry data is 'shared' across all instanced geometries, each instance have its own differentiating parameters such as translation, rotation, scaling, and physics. There are two types of instanced geometries:

• Animated instanced geometry: An animated instanced geometry is an instanced geometry that its translation, rotation or scaling changes over time.

• Non-animated instanced geometry: As its name implies, the translation, rotation, or scaling of a non-animated instanced geometry does not change over time.

Vanda Engine imports animated and non-animated instanced geometries from 3D tools capable of exporting to COLLADA.

Skeletal Animation

Skeletal animation is a technique in which a character is represented in two parts: a surface representation used to draw the character (called skin), and a hierarchical set of interconnected bones (called skeleton or rig) used to animate the skin.

Vanda Engine imports skins, skeletons, and animation clips (A collection of different animations) from 3D tools capable of exporting to COLLADA.

Lights

Vanda Engine imports animated and static directional, point, and spot lights from 3D tools capable of exporting to COLLADA.

Cameras

Vanda Engine imports animated and static cameras from 3D tools capable of exporting to COLLADA. See also <u>this section</u>.

External and built-in PhysX scenes

Each VScene contains *one* active PhysX scene. Vanda Engine supports 2 types of physics scenes:

- PhysX scenes imported from 3D tools capable of exporting to PhysX XML format.
- PhysX scenes generated inside Vanda Engine.

Built in physics system of Vanda Engine works with PhysX scenes generated inside Vanda Engine. Due to that, When you import an external PhysX scene, built in physics editor of Vanda Engine is disabled and you can not generate PhysX actors and triggers inside the editor. In order to generate the actors via built in physics editor, you should remove external PhysX scene. See also these sections:

- Introduction to PhysX
- <u>Physics Concepts</u>
- Physics Editor
- <u>PhysX XML format</u>
- <u>Triggers</u>

Built in Objects

Beside importing external scenes, Vanda Engine lets you create several built-in objects via the editor. These include:

- Lights
- Waters
- · Sounds
- · Sky Dome

Engine Objects Section describes these concepts in detail.

Extra Textures

Vanda Engine directly reads diffuse textures of geometries exported from 3D tools. Beside this functionality, Vanda Engine uses simple naming conventions to look for normal, alpha, shadow, dirt and gloss maps in diffuse texture path and assigns them to geometries. Moreover, you can assign or change these extra textures via the Material Editor.

See also:

- <u>Normal Map</u>
- <u>Gloss Map</u>
- <u>Alpha Map</u>
- <u>Shadow Map</u>
- Dirt Map
- <u>Diffuse Map</u>
- <u>Scene Options</u>

Rendering Effects

Bloom

After importing the instanced geometries or skins from external scenes, Vanda Engine automatically assigns a post-processing effect called bloom to the current VScene.

Dynamic Shadow

Vanda Engine uses a cutting-edge technology called Cascaded Shadow Maps (CSM) to render dynamic shadows. all of the instanced geometries cast dynamic shadows on other instanced geometries.

Fog

Vanda Engine has an exponential per vertex fog built-in.

Depth of Field

Vanda Engine supports depth of Field (DOF). DOF describes the distance in front of and behind a focus point that appears sharp in a photograph. By default, Fog and DOF are disabled.

You can modify the bloom, shadow, fog and depth of field properties via the Modify menu. See also:

- Introduction to CSM
- <u>Shadow Properties</u>
- Introduction to DOF
- <u>DOF Properties</u>
- Introduction to Fog
- Fog Properties
- Introduction to Bloom Effect
- <u>Bloom Properties</u>

Other Elements

Scene Manager

Vanda Engine uses a combination of Bounding Sphere and Octree to manage the instanced geometries and skins of the current VScene.

For more information about scene management in Vanda Engine, please visit

the following sections:

- Introduction to Scene Manager in Vanda Engine
- <u>Scene Manager Properties</u>

Scripting

You can assign scripts to instanced geometries and skins.

Vanda Engine 1.3.3 \rightarrow Vanda \rightarrow Learning the GUI	\leftarrow Previous page Next page \rightarrow
ODO: To change the footer's content go to Dr.Explain menu Options : Pro	piect Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setu** HTML Template and Layout

The contents of Learning the GUI

Main Menu
Button Bar
Left Bar
Viewports
Right Area
Bottom Bar

Made with help of Dr.Explain

)

$\frac{\text{Vanda Engine 1.3.3}}{\rightarrow} \rightarrow$ Main Menu	Vanda \rightarrow Learning the GUI	\leftarrow Previous page Next page \rightarrow
ODO. To change the footer's cor	ntent do to Dr Explain menu Ontions :	Project Settings · HTML (CHM) Export · Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Main Menu

- File Menu
- View Menu
- Create Menu
- Modify Menu
- **Tools Menu**
- Window Menu
- Help Menu

Made with help of Dr.Explain

 \mathbf{F}





This menu contains commands relating to the handling of files, such as open, save, etc:

New...

Closes the current VScene and creates a new empty VScene.

Open...

Opens a standard File dialog that lets you select an existing file. The file dialog, by default, only looks for files that end in ".vin".

Save

Saves the current VScene. For the first time, it opens a dialog that lets you select the file path and assigns a complex password to saved assets.

Save As...

Opens a File dialog that lets you save the scene under a new name.

Import

Lets you import external scenes in COLLADA (*.dae) or PhysX XML (*.xml) formats.

Publish

Opens a dialog that lets you publish your desired VScenes.

Exit

Closes the current VScene and exits the program.





The view menu contains the following:

Icons | Light Icons

This allows you to view the light icons.



Point Light Icon (*C* means external Light imported from 3D tools in COLLADA format)



Directional Light Icons (*C* means external Light imported from 3D tools in COLLADA format)



Spot Light Icons (*C* means external Light imported from 3D tools in COLLADA format)

Icons | *Sound Icons* This allows you to view the static sound icons.



Icons | Water Icons

This allows you to view the water icons.



Icons | Camera Icons

This allows you to view the camera icons.



Camera Icon

Icons | *Sky Icon* This allows you to view the sky icon.



Perspective Grids

This allows you to view the grids in the perspective viewport.

Octree

This allows you to view the octree partitioning in the perspective viewport. An octree is a tree data structure in which each internal node has exactly eight children. Vanda uses octree to partition the current scene by recursively subdividing it into eight octants. Octree in Vanda is used to boost the performance. See also <u>this section</u>.

Bounding Box

This allows you to view the object's bounding box in the perspective viewport.

A bounding box is a closed box that completely contains the skin or instanced geometry.

Bounding boxes in Vanda are used for frustum culling to boost the performance.



Bounding boxes are shown in white while Octree partitions are shown in pink or green.

Shadow Depth Map

In Vanda Engine 1, directional lights can use cascaded shadow depth buffers for whole scene shadows. In this technique, the camera view frustum is splitted and a separate depth map is created for each partition. This menu allows you to view separate depth map of each partition. See also <u>this section</u>

Polygon Mode

Allows you to select a polygon rasterization mode.

- *Point*: Polygon vertices that are marked as the start of a boundary edge are drawn as points
- *Line*: Boundary edges of the polygon are drawn as line segments.
- *Fill*: The interior of the polygon is filled.

Statistics

Shows the following information in the perspective viewport:

- · FPS (Frames Per Second)
- · Number of vertexes currently rendered in your VScene
- Number of triangles currently rendered in your VScene

Vanda Engine 1 uses a combination of octree and bounding box to only render the skins and instanced geometries that are inside the view frustum. Due to that, the above information may change based on the following parameters:

- current camera position
- current camera orientation
- current view frustum

Report

Displays a text window with useful information about everything that happens behind the scene.

Warnings and Errors

Displays a text window that shows the errors and warnings related to COLLADA files.

View | Layers

Each instanced geometry or skin may contain several texture layers. By default, VE combines these texture layers in order to create the final result. You can

watch each individual layer with the entries of this menu. It's useful for debug purposes.

See also these sections:

- Normal Map
- <u>Gloss Map</u>
- <u>Alpha Map</u>
- <u>Shadow Map</u>
- <u>Dirt Map</u>
- Diffuse Map



Create Menu



This menu contains all of the built-in objects of Vanda Engine 1:

Light

Opens a dialog that lets you select the properties of a new light. <u>This section</u> describes how to create a point, spot, or directional light.

Ambient Sound...

Opens a dialog that lets you select the properties of a new ambient sound. <u>This</u> <u>section</u> describes how to create an ambient sound. You are only allowed to insert one ambient sound in your scene.

Static Sound ...

Opens a dialog that lets you select the properties of a new static sound. <u>This</u> <u>section</u> describes how to create static sounds.

Water...

Opens a dialog that lets you select the properties of a new water object. <u>This</u> <u>section</u> describes how to create water objects.

Sky Dome...

Opens a dialog that lets you select the properties of a new sky dome. This section describes how to create a sky dome. You are only allowed to insert one sky dome in your scene.



Modify Menu



The Modify menu lets you modify all of the elements of Vanda Engine 1.

Shadow...

Opens a dialog that lets you change the dynamic shadow properties of the VScene. For more information about dynamic shadow and its properties in Vanda Engine 1, see <u>Introduction to CSM</u> and <u>Shadow Properties</u>.

Depth of Field...

Opens a dialog that lets you change the DOF properties of the VScene. For more information about DOF and its properties in Vanda Engine 1, see <u>Introduction to DOF</u> and <u>DOF Properties</u>.

Fog...

Opens a dialog that lets you change the fog properties of the VScene. For more information about fog and its properties in Vanda Engine 1, see <u>Introduction to</u> <u>fog</u> and <u>Fog Properties</u> sections.

Bloom...

Opens a dialog that lets you change the properties of bloom. For more information about bloom and its properties in Vanda Engine 1, see <u>Introduction</u> to <u>Bloom effect</u> and <u>Bloom properties</u> sections .

Light...

Opens a dialog that lets you change the general properties of light. For more information about general light properties in Vanda Engine 1, see <u>general light</u> <u>properties</u>.

Scene Manager...

Opens a dialog that lets you change the number of static instanced geometries per final octree split. For more information about scene manager in Vanda Engine 1, see <u>Scene Manager Properties</u>.

General PhysX Options...

Opens a dialog that lets you change the general properties of PhysX engine. For more information about physics properties, please visit <u>this section</u>.

Camera | Increase Speed

adds one unit to the speed of PhysX or Free camera .

Camera | Decrease Speed

Subtract one unit from speed of PhysX or Free camera .

Camera | Default Speed

Resets the PhysX or Free camera to its default speed.

Camera | Default Zoom

Resets the PhysX or Free camera to its default zoom.

Geometry | Cull Faces (Selected Geometry)

Only works in geometry based selection. If you select a polygon mesh of an instanced geometry or skin, this menu allows you to enable or disable its back-face culling. By default, back-face culling is enabled for all of the skins and instanced geometries. This default behavior makes rendering instanced geometries and skins quicker and more efficient by reducing the number of polygons for the program to draw.

Geometry | Materials...

Opens a dialog that lets you change the materials of skins and instanced geometries. "Modify Menu in Depth" section describes how to modify the properties of the selected skins and instanced geometries.

Geometry | Physics...

Opens a dialog that lets you generate the PhysX actors for the selected skin and instanced geometry. "Modify Menu in Depth" section describes how to generate the PhysX actors for the selected skins and instanced geometries.

Geometry | Scripts...

Opens a dialog that lets you assign the files containing Lua scripts to the enter and exit events of the triggers. See <u>Script Manager</u>.

Geometry | Modify Geometry (All instanced Geometries)

Lets you specify an ambient color for all of the instanced geometries and skins of current VScene. It's useful for external scenes that don't contain ambient color. For example, Cinema 4D does not export ambient colors for instanced geometries.

Geometry | Geometry based selection

If checked, your selection is done based on the polygon mesh (all of the vertices, edges and triangles) of your instanced geometries. Otherwise, Material Based Selection is activated which allows you to select the instanced geometries based on their materials.



Tools Menu



The Tools menu simply provides menu access to all of the tools located in Vanda Engine 1.

Camera and Rendering...

Opens a dialog that lets you select and activate a camera from the list. By default, there are two camera types in your scene. However if you import any other cameras via Import | COLLADA menu, they are shown in this dialog.

We are planning to support a feature that lets you render the current VScene from the perspective of activated camera. If activated camera contains animations, Vanda Engine will capture a video in AVI format. Otherwise it will render the activated camera view in a TGA file.

Water Attachment...

By default, no instanced geometry or skin in your scene is attached to any of the water objects. This menu allows you to attach the selected instanced geometry or skin to a desired water object. To attach an instanced geometry or skin, you should select it in your scene (in geometry mode selection) and click on this menu entry. If any water objects exist in your scene, this menu opens a dialog that lets you select a water object from the list. To attach your selected instanced geometry or skin to the selected water object, simply click the Attach button in this dialog.

General Options...

Opens a dialog that lets you set the general options. See <u>General Options</u>.

Scene Options...

Opens a dialog that lets you set the scene options. See <u>Scene Options</u>.


Window Menu



This menu lets you switch between the perspective viewport and 4 sided viewport.



Help Menu Co Co Co

This menu provides help. Where to get help if you are stuck.

Online Help...

Lets you access this help file.

About...

About this program.

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Button Bar

Button Bar

Made with help of Dr.Explain

•



Button Bar



A Button bar is a strip of icons used to perform frequently used functions.



Closes the current VScene and creates a new empty VScene.



Opens a standard File dialog that lets you select an existing file. The file dialog, by default, only looks for files that end in ".vin".



Opens a standard File dialog that lets you import an external scene in COLLADA format. Such files end in ".dae".



Opens a standard File dialog that lets you import an external PhysX scene in XML format.



Saves the current VScene.



Opens a File dialog that lets you save the VScene under a new name.



Lets you publish your desired VScenes.



Lets you access to Camera and Rendering dialog.



Opens a dialog that lets you assign a script to the selected instanced geometry or skin.



Activates the previous animation of the selected animated instanced geometry or skin. Useful for debug mode.



Activates the next animation of the selected animated instanced geometry or skin. Useful for debug mode.



Plays current animation of the selected animated instanced geometry or skin. Useful for debug mode.



Pauses current animation of the selected animated instanced geometry or skin. Useful for debug mode.



Lets you enable or disable timer.



Opens a dialog that lets you change the properties of the selected instanced geometry or skin.



Opens a dialog that lets you manage the PhysX actors of the selected instanced geometry or skin.



Opens the official YouTube page of Vanda Engine.



Opens the official Twitter page of Vanda Engine.



Opens the official Facebook page of Vanda Engine.



Opens the official website of Vanda Engine.

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Learning the C</u> → Left Bar	GUI ← Previous page Next page →
ODO: To change the footer's content go to Dr.Explain menu On t	tions : Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Left Bar

Left Bar

Made with help of Dr.Explain

 \rightarrow





A Left bar is a vertical strip of icons used to perform frequently used functions. This bar usually refers to creating the Vanda Engine 1 objects.



Opens a dialog that lets you select the properties of a new sky dome. This section describes how to create a sky dome. You are only allowed to insert one sky dome in your scene.



Opens a dialog that lets you select the properties of a new water object. <u>This</u> <u>section</u> describes how to create water objects.



Opens a dialog that lets you select the properties of a new light. <u>This section</u> describes how to create a point, spot, or directional light.



Opens a dialog that lets you select the properties of a new static sound. <u>This</u> <u>section</u> describes how to create static sounds.



Opens a dialog that lets you select the properties of a new ambient sound. <u>This</u> <u>section</u> describes how to create an ambient sound. You are only allowed to insert one ambient sound in your scene.



Saves the position and orientation of the Free or PhysX camera.



Displays a text window with useful information about everything that happens behind the scene



Displays a text window that shows the errors and warnings related to COLLADA files.

Vanda Engine 1.3.3 → Viewports	\rightarrow Vanda \rightarrow Learning the C	GUI <u>←</u> Previous p	bage <u>Next page</u> →
ODO: To change the footer's	content do to Dr Evolain menu Ont	ions · Project Settings · HTM	(CHM) Export · Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Viewports

- Introduction to viewports
- Perspective Viewport
- Orthographic Viewports

Made with help of Dr.Explain

=)



HTML Template and Layout

Introduction to viewports



Vanda Engine 1 needs a window which it will render into. Viewports are rectangular areas within the window.

The standard viewports of Vanda Engine 1 show you four different views of your current scene: perspective, front, top and side view.



Perspective viewport (Lower left) uses perspective cameras, while orthographic viewports (Top, Side, Front) use orthographic camera.

Orthographic cameras do not show perspective. Perspective cameras distort the view the way a real-world camera would.

At times you might want to look at your scene through a large, undivided viewport, giving you a "picture-window" view of your scene. To do this, you can use *Window* | *Switch* from the main menu.



Only one viewport can be in the active state at a time. When working with multiple viewports, the last view you clicked in is the active view. The active viewport is indicated by a yellow border.



Perspective Viewport



Coordinate System

Vanda Engine 1 uses the right-handed Cartesian Coordinate System. The X (red), Y (green) and Z (blue) axis are perpendicular and range from -infinity to +infinity on each axis. The positive X is to the right, the positive Y goes up and the positive Z goes forward.



Right-handed Cartesian Coordinate System in Vanda Engine 1

Free Camera

By default, Vanda Engine 1's perspective viewport uses a Free camera. This camera does not interact with physics actors or triggers and lets you move and rotate around the current VScene.

Navigating the Free Camera

Use the following controls to navigate the default camera in your perspective viewport:

- **Left Mouse Button**: Drag to orbit the camera left, right, up and down.
- **Right Mouse Button**: Drag to zoom in and out.

You should hold the following keys *while* holding the Left Mouse Button to move the camera in your VScene. Note that forward direction of your movement is determined by the front direction of default camera. Other directions are determined relative to the forward direction.

- **W Key**: Hold to move forward.
- **S Key**: Hold to move backward.
- **A Key**: Hold to slide left.
- **D Key**: Hold to slide right.
- **Q Key**: Hold to move up (It does not work if you enable gravity).
- **E Key**: Hold to move down.

Alternatively, you can push middle mouse button once to activate the perspective viewport. In order to deactivate the perspective viewport, push middle mouse button again.

3rd Person Controller System

Vanda Engine 1's perspective viewport uses a 3rd Person Controller system as well. This system consists of a PhysX capsule character controller and a

perspective camera attached to it.

Character controller in Vanda Engine 1 acts as a collider and mainly used for character movement, while still letting it be affected by the game environment, i.e., sliding over surfaces, along walls, etc.

By default, there is no gravity applied to the PhysX controller and there is no PhysX static or dynamic actor in your scene. So by default, the character controller does not fall and is not affected by the environment.

If you enable gravity, the controller will fall onto and rest on a static rigid body (If there is any, otherwise it continues to fall until you disable gravity).

Fore more information about PhysX character controller in Vanda Engine 1, please <u>visit this section</u>.

The perspective camera attached to the controller follows its movements. It is supposed to be looking always to the head of character controller. So rotating with left mouse button orbits the camera around the head of character controller. This camera can interact with static and dynamic actors in your scene to prevent scene clipping.



Navigating the 3rd Person Controller System

Use the following controls to navigate the default camera in your perspective viewport:

- **Left Mouse Button**: Drag to orbit the camera left, right, up and down.
- **Right Mouse Button**: Drag to zoom in and out.

You should hold the following keys *while* holding the Left Mouse Button to move the character controller and camera in your scene. Note that forward direction of your movement is determined by the front direction of default perspective camera. Other directions are determined relative to the forward direction.

W Key: Hold to move forward.

- **S Key**: Hold to move backward.
- **A Key**: Hold to slide left.
- **D Key**: Hold to slide right.
- **Q Key**: Hold to move up (It does not work if you enable gravity).
 - **E Key**: Hold to move down.

When you are moving with the above keys, you can simultaneously drag Left Mouse Button to orbit the camera around character controller and change the direction of your movement.

Alternatively, you can push middle mouse button once to activate the perspective viewport. In order to deactivate the perspective viewport, push middle mouse button again.

Active camera

The active camera is the camera that is currently used for rendering the scene. By default, the default Free camera is active. Only one camera can be in active state at a time. To activate a camera, go to Tools | Camera | Camera and Rendering... and in the new dialog that appears, select the camera you would like to make active. By default, external cameras are not active in perspective viewport. If you activate an external camera, its animation (if exists) is activated and it renders the scene from its point of view. However you have no control over its movement, zoom, etc. Due to that, external cameras are suitable for cut-scenes and capturing the videos. If you would like to navigate the camera and character controller in your perspective viewport, make sure that the Free or 3rd person camera is active.

We are planning to add a feature that lets you capture the videos and images from the perspective of current activel camera.

Icons

external lights and camera objects and each object created inside Vanda Engine 1 have an associated icon inside perspective viewport. These icons have three purposes:

- Show object position
- Show object type
- Help you move the object by selecting and moving its icon

You are not allowed to move the external lights and cameras.



Icons in this image show three different objects: A point light, a sky dome, and a static sound. As you see, Static sound's icon has been selected.

Selection

To select 3D instanced geometries or skins and edit the textures or of their geometries, follow these instructions:

• There are two types of selection for 3D instanced geometries or skins: 1. Geometry based selection, and 2. Material based selection. Default selection method is geometry based selection which means that your selection is done based on the polygon meshes of your instanced geometries or skins. Select the method that suits your needs from the main menu.

• Left click on the instanced geometry or skin that you want to view/change its geometry textures. The selected instanced geometry is shown in pink and its name appears in the Name area which is located at the lower left of the editor.

• The Map area is activated and shows the associated textures of the geometry of selected instanced geometry or skin and lets you manage its textures.

Although selection is based on *instances of geometries*, textures are assigned to *geometries* (not to instanced geometries). Vanda Engine assumes that texture data relates to the geometry of currently selected instance or skin and is shared among all of the instances of that geometry.

To edit the physics attributes of instanced geometries or skins, select your desired instanced geometry and go to Skin and instanced Geometry... menu. PhysX section of the Skin and instanced Geometry dialog lets you assign physics actors and triggers to instanced geometries and skins.

PhysX data is assigned to instanced geometries and skins.

To select and translate Engine Objects such as lights, sounds, sky and so on, follow these instructions:

· Left click on the icon of your desired object. The selected icon is shown in red and the red (x), green (y) and blue (z) handles appear.

• Select red, green, or blue handles and drag Left Mouse Button to translate



Orthographic Viewports



Orthographic viewports use orthographic camera.

An orthographic camera in 2D viewport views the scene by looking straight down one of the world axes: X, Y, or Z. The Front camera looks straight down the Z axis. The side camera looks straight down the X axis. The Top camera looks straight down the Y axis.

You also have pan and zoom features available in orthographic viewports.

Navigation

- **Left Mouse Button**: Drag to move the camera up, down, left or right.
- **Right Mouse Button**: Drag to zoom in or out.

Alternatively, you can push middle mouse button once to activate the orthographic viewports. In order to deactivate the orthographic viewports, push middle mouse button again.

<u>Vanda Engine 1.3.3</u> → <u>V</u> → Right Area	← Previous page Next page →

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Right Area

- External Scenes
- External Elements
- Engine Objects
- PhysX Elements

Made with help of Dr.Explain

)



External scenes



This area lists all external scenes that are imported successfully by Vanda Engine 1.



External Scenes area

Checkbox

Each external scene is rendered in its own layer. The checkbox specifies whether the external scene should be rendered. For example, sometimes you need to work on specific scenes to generate their colliders. In this case you can disable other layers and easily work on the visible layers. Another example: Sometime you need to disable an external scene and show it while colliding with a trigger.

When you save the current VScene, the checkbox status of each layer is saved a well. So if you publish the VScene, all of the external scenes marked as invisible are not rendered. So before publishing the VScene, make sure that all of the checkboxes are checked correctly.

To delete an imported external scene , follow these instructions:

- Left-click the name of the external scene that you want to remove from Vanda Engine 1.
- Click the \times button.
- In a dialog that appears, click Yes button.

The delete operation does not remove the textures shared among other external scenes.



HTML Template and Layout

External Elements



This area lists all of the external elements supported by Vanda Engine 1 that have been imported successfully.



External Elements area

Here's the list of the external elements that are supported by Vanda Engine 1:

- 🎯 Mesh
- 🔊 Camera
- **fx** Effect
- 🛃 Image
- 🔛 Material
- 📔 Light
- 🖉 🕅 Animation and animation clip
- · Skin Controller



HTML Template and Layout

Engine Objects



This area lists all of the built-in objects of your VScene.

<i>«</i> ×

Engine Objects area

Here's the list of the objects that are supported by Vanda Engine 1:

- Sky Dome
- Light
- The Ambient Sound
- **Static Sound**
- **Water**



HTML Template and Layout

PhysX Elements



This area lists all of the PhysX elements of your scene that are supported by Vanda Engine 1.



PhysX Elements

Here's the list of the PhysX elements that are supported by Vanda Engine 1:

- E Static Rigid Body
- Dynamic Rigid Body
- Trigger

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>L</u> → Bottom Bar	earning the GUI	← Previous page <u>Next page</u> →
ODO. To change the footer's content go to Dr E	xplain menu Options · Project	Settings · HTML (CHM) Export · Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Bottom Bar

Bottom Bar

Made with help of Dr.Explain

 \rightarrow



TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout



Bottom Bar contains 4 textual areas:

- 1. Shows the last line of the Report window.
- 2. Shows the last line of the Warnings and Errors window.
- 3. Shows the status (enabled or disabled) of important features.

4. Shows the name of the selected polygon mesh or material of instanced geometry or skin.
| <u>Vanda Engine 1.3.3</u> → <u>Vanda</u>
→ Modify Menu in Depth | ← Previous page Next page → | | | |
|--|-----------------------------|--|--|--|
| | | | | |
| TODO: To change the footer's content go to Dr.Explain menu Options : Project Settings : HTML (CHM) Export : Setup
HTML Template and Layout | | | | |

The contents of Modify Menu in Depth

~	Material Editor
,	Physics
,	Shadow
,	Depth of Field
,	Fog
,	Bloom
,	Light
,	Scene Manager
,	Script Manager

Made with help of Dr.Explain

)

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Modify Menu in Depth</u> → Material Editor	← Previous page Next page →
TODO: To change the footer's content go to Dr.Explain menu Options :	: Project Settings : HTML (CHM) Export : Setup

HTML Template and Layout

The contents of Material Editor

- Introduction
- Normal Map
- Gloss Map
- Alpha Map
- Shadow Map
- Dirt Map
- Diffuse Map

Made with help of Dr.Explain

▶



Introduction



This area is responsible for management of materials in Vanda Engine 1 and lets you assign Normal Map, Gloss Map, Alpha Map, Shadow Map, Dirt Map, Diffuse Map, Bias and Scale factors to the selected geometry or skin.



To access the Material Editor, you should follow these steps:

1. Select your desired instanced geometry or skin.

2. Click the Button. Alternatively, you can go to the Modify | Geometry | Material... menu.

Material Editor is relative to skins and instanced geometries. So if you don't select the skin or instanced geometry, you can't access this editor.

Although selection is based on *instances of geometries*, textures are assigned to *geometries* (not to instanced geometries). Vanda Engine assumes that texture data relates to the geometry of currently selected instance or skin and is shared among all of the instances of that geometry.



Normal Map



In Vanda Engine 1, a Normal Map is used to fake high-res geometry detail when it's mapped onto a low-res mesh. The pixels of the normal map each store a normal, a vector that describes the surface slope of the original high-res mesh at that point. The red, green, and blue channels of the normal map are used to control the direction of each pixel's normal.



A sample Normal Map

To apply a Normal Map to a skin or geometry, follow these steps:

• Select the instanced geometry or skin you want to apply Normal Map to its geometry.

- Click the 😎 button or go to Modify | Geometry | Material... menu.
- Click the Normal... button. This opens a standard File dialog that lets you select a DDS file.

• Select your desired Normal Map and Click Open.

To remove a Normal Map from a skin or geometry, follow these steps:

- Select the instanced geometry or skin.
- Click the 🕑 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Normal button.
- In a new dialog box that appears, click Yes button.



Gloss Map



Specular Gloss Map in Vanda Engine 1 is a a texture that controls the shape/power of the specular highlights from real-time lights. The higher the value of a pixel (from black to white), the shinier the surface will appear ingame. Therefore, surfaces such as dry stone or cotton fabric would tend to have a very dark specular map, while surfaces like polished chrome or plastic would tend to have lighter specular maps.



Sample Gloss Map

To apply a Specular Gloss Map to a geometry or skin, follow these steps:

 \cdot Select the skin or instanced geometry you want to apply Gloss Map to its geometry.

Click the Ӱ button or go to Modify | Geometry | Material... menu.

- Click the Gloss... button. This opens a standard File dialog that lets you select a DDS file.
- Select your desired Gloss Map and Click Open.

To remove a Specular Gloss Map from a geometry or skin, follow these steps:

- Select the instanced geometry or skin.
- Click the 😟 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Gloss button.
- In a new dialog box that appears, click Yes button.







Alpha map in Vanda Engine 1 is a texture that is applied to a 3D object and designates certain areas of the object to be discarded. So the alpha texture in Vanda Engine 1 is a black & white image containing information about where a specific object exists and where it does not.



Sample Alpha Map

To apply an Alpha Map to a geometry or skin, follow these steps:

- \cdot Select the skin or instanced geometry you want to apply Alpha Map to its geometry.
- Click the 😎 button or go to Modify | Geometry | Material... menu.
- Click the Alpha... button. This opens a standard File dialog that lets you

select a DDS file.

• Select your desired Alpha Map and Click Open.

To remove an Alpha Map from a skin or geometry, follow these steps:

- Select the skin or instanced geometry.
- Click the 🕏 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Alpha button.
- In a new dialog box that appears, click Yes button.



Shadow Map



Shadow Map in Vanda Engine 1 is kind of Lightmap, which is a grayscale texture that contains the brightness of surfaces. Therefore, a Shadow Map in Vanda Engine 1 is pre-computed, and may be used for static instanced geometries only. Pixel values of 1 don't affect surface brightness, while pixel values of 0 completely darken the surface. Shadow Mapping in Vanda Engine 1 uses hardware multitexturing to combine a regular diffuse map and dirt map -If they exist- with a shadow map. Vanda Engine 1 maps the Shadow Map to the 3D object based on its 2nd UV set.





Shadow Map. (A) Left image shows the scene with Shadow Map. (B) Right image shows the same scene without Shadow Map.

To apply a Shadow Map to a skin or geometry, follow these steps:

- Select the geometry you want to apply Shadow Map to its geometry. Now if your object contains 2nd UV set, Shadow... button in Material Editor is activated.
- Click the 💟 button or go to Modify | Geometry | Material... menu.
- Click the Shadow... button. This opens a standard File dialog that lets you select a DDS file.
- Select your desired Shadow Map and Click Open.

To remove a Shadow Map from a skin or geometry, follow these steps:

- Select the instanced geometry or skin.
- Click the 💟 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Shadow button.
- In a new dialog box that appears, click Yes button.



Dirt Map



A Dirt Map in Vanda Engine 1 defines the "noise" of a surface, such as that representing fine grain. Dirt Mapping in Vanda Engine 1 uses hardware multitexturing to combine a regular diffuse map and shadow map -If they existwith a dirt map. Vanda Engine 1 maps the Dirt Map to the 3D object based on its 2nd UV set.

To apply a Dirt Map to a skin or geometry, follow these steps:

- \cdot Select the instanced geometry or skin you want to apply Dirt Map to to its geometry.
- Click the 😎 button or go to Modify | Geometry | Material... menu.
- Now if your object contains 2nd UV set, Dirt... button in Material Editor is activated.
- Click the Dirt... button. This opens a standard File dialog that lets you select a DDS file.
- Select your desired Dirt Map and Click Open.

To remove a Dirt Map from a skin or geometry, follow these steps:

- · Select the skin or instanced geometry.
- Click the 😎 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Dirt button.

In a new dialog box that appears, click Yes button.





In Vanda Engine 1, a diffuse map is a texture you use to define a surface's main color. So in Vanda Engine, a diffuse and color map are the exact same thing.



Diffuse Map. (A) Left image shows the VScene without Diffuse Map. (B) Right image shows the same VScene with Diffuse Map applied to all instanced geometries.

To apply a Diffuse Map to a skin or geometry, follow these steps:

- Select the skin or instanced geometry you want to apply Diffuse Map to its geometry. Now the Diffuse... button in Material Editor is activated.
- Click the 😎 button or go to Modify | Geometry | Material... menu.
- Click the Diffuse... button. This opens a standard File dialog that lets you select a DDS file.
- Select your desired Diffuse Map and Click Open.

To remove a Diffuse Map from a skin or geometry, follow these steps:

- Select the skin or instanced geometry.
- Click the 📀 button or go to Modify | Geometry | Material... menu.
- Now select the button next to the Diffuse button.
- In a new dialog box that appears, click Yes button.



HTML Template and Layout

The contents of Physics

- Physics Editor
- General PhysX Options

Made with help of Dr.Explain

- **)**



Physics



Please refer to **Physics Editor**.



Options



Please refer to <u>this section</u>.

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Modify Menu in Depth</u> → Shadow	\leftarrow Previous page Next page \rightarrow
TODO: To change the footer's content go to Dr.Explain menu Option	ns : Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Set HTML Template and Layout**

The contents of Shadow

- Introduction to CSM
- Shadow Properties

Made with help of Dr.Explain

- **)**



Introduction to CSM



Shadow maps are a very popular technique to obtain realistic shadows in game engines.

When trying to use them for large spaces, shadow maps get harder to tune and will be more prone to exhibit surface acne and aliasing. Cascaded Shadow Map (CSM) is a known approach that helps to fix the aliasing problem by providing higher resolution of the depth texture near the viewer and lower resolution far away. This is done by splitting the camera view frustum and creating a separate depth-map for each partition in an attempt to make the screen error constant.

Capturing everything in a single shadow map would require very high and impractical resolution. Thus, several shadow maps are used - a shadow map that covers only nearby objects so that each casts a detailed shadow; another shadow map that captures everything in the distance with coarse resolution and optionally some more shadow maps in between. This partitioning is reasonable because objects that are far away cast shadows that in screen space occupy just a few pixels and close-by objects might cast shadows that occupy a significant part of the screen.



CSM splits the camera view frustum and creates a separate depth-map for each partition



Result of CSM

Vanda Engine 1 uses Parallel split CSM technique and only one "directional" light can cast shadows on skins and instanced geometries.

For more information about the CSM technique, please refer to <u>this</u> article.





To open the Shadow Properties dialog, choose Modify | Shadow... from the main menu.



Shadow Properties Dialog

Rendering Algorithm

Lets you select one of the following rendering methods to render dynamic

shadows:

Debug: Lets you view the splits.

- Simple
- Leak
- No leak
- · PCF
- PCF Trilinear
- · PCF 4 Tap
- PCF 8 Tap
- PCF Guassian

Some algorithms such as PCF Guassian or PCF 8 Tap generate better quality. But they require more computation.

Number of Splits

This option lets you specify the number of view frustum splits.

Although more splits give you better results, they require more computation and are more expensive.

Weight of Splits

Lets you specify weight of splits. A value of 0 creates uniform splits, while larger weight values split the view frustum exponentially-For example subsequent cascades can have near...far distances like: 0.5 ... 7, 7 ... 25, 25 ... 100.

Near Clip Plane

Lets you specify the near clip plane of view frustum. This plane is positioned relative to the location of current active camera.

Far Clip Plane

Lets you specify the far clip plane of view frustum. This plane is positioned relative to the location of current active camera.

Near and Far Clip Planes in Shadow Properties dialog are solely used for computing the dynamic shadow and don't change the properties of main and external cameras.

Resolution

Allows you to select one of the following resolutions for shadow depth buffer:

1024 * 1024 2048 * 2048 4096 * 4096

Although larger shadow depth buffers give you better results, they require more computation and are more expensive.

Intensity

Lets you specify the intensity of shadow. A value of 1 results in black shadows, while a value of 0 removes shadow color completely.

Directional Light

Lets you choose one of the directional lights of your scene to cast dynamic shadows.

If you don't create any directional lights in your scene, instanced geometries or skins don't cast shadows.

Enable Dynamic Shadow

Lets you enable or disable whole scene dynamic shadows.

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Modify Menu in Depth</u> → Depth of Field	\leftarrow Previous page Next page \rightarrow
TODO: To change the footer's content go to Dr.Explain menu Options : F	Project Settings : HTML (CHM) Export : Setup

HTML Template and Layout

The contents of Depth of Field

- Introduction to DOF
- **DOF** Properties

Made with help of Dr.Explain

•



Introduction to DOF



Broadly, the term depth of field (DOF) describes the distance in front of and behind a focus point that appears sharp in a photograph.

Focus Point in Vanda Engine 1 is the main point of focus that stands out from everything else. Focus Range in Vanda Engine 1 is the distance in front of and behind the Focus Point. Any instanced geometry or skin that stands in between [Focus Point + Focus Range] and [Focus Point - Focus Range] appears sharp in your scene. The depth of field in Vanda Engine 1 does not abruptly change from sharp to unsharp, but instead occurs as a gradual transition. In fact, everything immediately in front of or in back of the focusing distance begins to lose sharpness.



DOF. (A) and (C) instanced geometries are blurred, while (B) instanced geometries are sharp



DOF Properties



To access the DOF properties dialog, choose Modify | Depth of Field... from the main menu.

Focus Point

Lets you set the focus point.

Focus Range Lets you set the focus range

Enable Depth of Field

Lets you enable or disable depth of field effect. If you disable DOF, everything in your scene will be sharp.

Debug

As its name suggests, it's used for debugging the DOF. When checked, skins and instanced geometries that stand in between [Focus Point + Focus Range] and [Focus Point - Focus Range] appear visible, while fully blurred objects appear invisible (white). Since depth of field in Vanda Engine 1 does not abruptly change from sharp to unsharp, objects that stand outside [Focus Point + Focus Range] and [Focus Point - Focus Range] lose their visibility gradually (From original color to white color).


DOF debug mode in Vanda Engine 1

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Modify Menu in Depth</u> → Fog	← Previous page Next page →
TODO: To change the footer's content go to Dr.Explain menu Option	s : Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setu** HTML Template and Layout

The contents of Fog

- Introduction to fog
- **Fog Properties**

Made with help of Dr.Explain

•



Introduction to fog



To compute the fog color, we should use an equation that allows us to define the way fog density behaves according to the distance between the camera and a given vertex. The aim of this equation is to compute a fog factor that will allow to mix the fragment color with the fog color. Vanda Engine 1 uses exponential method, because of its more realistic fog rendering possibilities.

 $fogFactor = e^{(-(density * z)^2)}$

z is the distance between the camera and the currently processed vertex.







To access Fog properties dialog, choose Modify | Fog... from the main menu.

Select

Lets you change the fog color

Density

This option Lets you change the density factor.

Enable Fog

This checkbox lets you enable or disable whole scene fog.



HTML Template and Layout

The contents of Bloom

- Introduction to Bloom Effect
- Bloom Properties

Made with help of Dr.Explain

 \rightarrow

Vanda Engine 1.3.3 \rightarrow Vanda	\leftarrow Previous page Next page \rightarrow	
\rightarrow Modify Menu in Depth \rightarrow Bloom		
→ Introduction to Bloom Effect		
TODO: To change the factor's content go to Dr Evalain many Ont	ions - Brojost Sottings - HTML (CHM) Export - Sotup	
TODO: To change the footer's content go to Dr.Explain menu Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout		

Introduction to Bloom Effect



The physical basis of bloom is that, in the real world, lenses can never focus perfectly. Under normal circumstances, these imperfections are not noticeable, but an intensely bright light source will cause the imperfections to become visible. As a result, the image of the bright light appears to bleed beyond its natural borders. Bloom Lighting is a commonly used post-processing effect. It gives shiny halos of light to particularly bright parts of objects, as is seen with most camera lenses.



Bloom effect in Vanda Engine 1



The same scene without Bloom effect





To access Bloom Properties dialog, choose Modify | Bloom... from the main menu.

Color

Lets you change the color of bloom effect.



Vanda Engine 1 lets you change the color of bloom

Intensity

Lets you change the intensity of bloom effect. accepted values are in the range [0, 1]. A value of 0 disables bloom effect.

Enable Bloom

Lets you enable or disable bloom effect.







To access the general light properties, choose Modify | Light... from the main menu.

COLLADA Lights

COLLLADA lights have a color associated to them. Vanda Engine 1 assigns this color to their diffuse Channel, but it doesn't know anything about the Ambient and Specular channels and shininess of COLLADA lights. This section lets you set these properties for all COLLADA lights.

Ambient

Lets you assign the Ambient color of COLLADA lights.

Specular

Lets you assign the Specular color of COLLADA lights.

Shininess

Lets you assign the shininess property of COLLADA lights.

These settings are assigned to all COLLADA lights. Dues to that, it's recommended to use Engine Lights instead of COLLADA lights. Engine lights give you more control over their properties.

Engine and COLLADA Lights

Tolerance

Instanced geometries and skins that lie inside light radius are affected by that light.

If quadratic attenuation is greater than 0, this property affects light radius. Higher tolerance values decrease light radius, while lower values increase it.

Vanda Engine 1.3.3 \rightarrow Vanda \rightarrow Modify Menu in Depth \rightarrow Scene Manager	← Previous page Next page →
TODO: To change the footer's content go to Dr.Explain menu Options :	Project Settings : HTML (CHM) Export : Setup

HTML Template and Layout

The contents of Scene Manager

- Introduction to Scene Manager in Vanda
- Scene Manager Properties

Made with help of Dr.Explain

 \rightarrow



Introduction to Scene Manager in Vanda



Overview

The view frustum is the volume that contains everything that is potentially (there may be occlusions) visible on the screen. This volume is defined according to the camera's settings, and when using a perspective projection takes the shape of a truncated pyramid.

The apex of the pyramid is the camera position and the base of the pyramid is the far plane. The pyramid is truncated at the near plane, hence the name frustum.



All the stuff that will potentially be visible on screen is inside, at least partially, the truncated pyramid, so there is no need to try and render what is outside the frustum, since it won't be visible anyway.



In the figure above all the green stuff (totally inside the view frustum) and all the yellow stuff (partially inside) would be rendered, whereas the red stuff will not be rendered.

Scene Manager in Vanda Engine

In Vanda Engine, a scene manager improves the frame rate by rendering only the *visible* instanced geometries and skins of the current VScene.

Vanda Engine uses a combination of *Octree* and *Bounding Box* to manage static instanced geometries. Moreover, Vanda Engine simply uses Bounding Box to detect visible dynamic instanced geometries (Animated or controlled by Physics engine) and skins.

Bounding Box

Vanda Engine puts bounding box around each instanced geometry or skin and just calculates if the vertices of bounding box are viewable. If one or more vertices of the bounding box are visible, Vanda Engine renders the instanced geometry or skin.

Octree

When there are lots of static instanced geometries, It's expensive to put bounding boxed around all of the static instanced geometries and calculate if those boxes are viewable. To use less bounding boxes, Vanda Engine uses a technique called Octree.

An Octree is a tree data structure in which each internal node has exactly eight children. An Octree system in Vanda Engine recursively splits the VScene into eight regions and only renders the content of regions that are in the view area of the camera.



But when should we start and stop dividing octree cubes into smaller ones? It's specified by a property that specifies the number of objects per *final* octree cubes. For example assume that each final Octree cube should contain 2 static instanced geometries and your VScene contains 30 static instanced geometries. At the beginning, our Octree contains one cube with 30 instanced geometries inside it. Since it contain more than 2 instanced geometries, it's not a final cube. Due to that, Vanda Engine *attaches* all of the instanced geometries to the first Octree cube and divides the first cube to eight smaller cubes. It repeats this process for smaller cubes until each final cube contains 2 or less static instanced geometries. For example if a child cube named "CubeLevel2" contains 10 instance geometries, Vanda Engine attaches these 10 instanced geometries to CubeLevel2 and divides CubeLevel2 to 8 smaller cubes and repeats the process for smaller cubes.



The most important things to understand about Octrees is that if a cubic parent of Octree is not inside the view frustum, then its children aren't either; and if a cubic parent of Octree is completely inside the view frustum, then its children are either. So Vanda Engine doesn't need to put bounding boxes around the static instanced geometries of invisible or completely visible Octree cubes to calculate if those spheres are viewable.

Each parent Octree cube contains all static instanced geometries of its child cubes.





To access the scene manager dialog, go to Modify | Scene Manager... menu. This will open a dialog that lets you change the properties of scene manager.

Number of Static Instanced Geometries Per Final Splits

Lets you specify the number of static instanced geometries per final cube of octree.



Script Manager



To access the Script Manager dialog, select a skin o instanced geometry marked as trigger in the VScene and choose Modify | Geometry | Script... from the main menu. This dialog lets you assign scripts to trigger events. Alternatively,

you can click the 💟 button.

VandaEngine RTI Pro Version 1.2.0		- 6 8
File View Create Modify Physik Tools Window Help	0 🕀 🕲 🔗 🦻	🥪 🕃 🗊 😒
First First		External Scenes
		box.DAE
2		2 ×
	Script Manager	External Elements
((C))	Trgger::Enter Remove OK	fX ColorEffect
77	Trigger:Ext Remove Cancel	ColorEffect PEnvironment
1		😚 👘 👘 👘 👘 🖓 geom - Box 01
		Engine Objects
A		
		PhysX Elements
		PhysX_mesh_25
E CONTRACTOR		
		×
In 105> Ready	Ln 2> box.DAE - 0 error (s), 2 warning (s)	PhysX Debug mode enabled geom-Box01

Script Manager

Trigger::Enter

Opens a dialog that lets you select a lua script (*.lua) for enter event of selected trigger.

Trigger::Exit

Opens a dialog that lets you select a lua script (*.lua) for exit event of selected trigger

Report

Reports the errors of imported scripts.

For more information about scripting in Vanda Engine, please visit <u>Introduction</u> to Lua and <u>functions</u>.

Vanda Engine 1.3.3 \rightarrow Vanda \rightarrow Engine Objects	← Previous page <u>Next page</u> →
	2
ODO: To change the footer's content go to Dr.Explain menu Options : Pr	oject Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu Options : Project Settings : HTML (CHM) Export : Setu HTML Template and Layout

The contents of Engine Objects

Lights Sounds . **Water** Sky Dome

Made with help of Dr.Explain

 \rightarrow



TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**



To access the Light Editor, choose Create | Light... menu.



In This photo: A blue directional light, a green spot light, and a red point light.

Name

Lets you choose a unique name for this light

Light Type

Vanda Engine 1 supports three types of lights: directional light, point Light, and spot light.

• **Directional** lights affect everything in the scene, like the sun. Their position is irrelevant.

• **Point** lights shine from a location equally in all directions, like a light bulb.

• **Spot** lights shine from a point in a direction and only illuminate skins and instanced geometries within a cone - like the headlights of a car.

Ambient

Lets you specify the color of ambient property of the light. Ambient lighting is a general illumination that comes from all directions and equally brightens the scene.

Diffuse

Lets you specify the color of diffuse property of the light. The main difference between ambient light and diffuse light is the fact that diffuse light is dependent on the direction of the rays of light while ambient light ignores it completely. When only ambient light is present the entire scene is equally lit. Diffuse light makes the parts of instanced geometries and skins that face the light brighter than the parts that are opposite from it.

Specular

Lets you specify the color of specular property of the light. A specular highlight is the bright spot of light that appears on shiny instanced geometries and skins when illuminated.

Position/Direction

Lets you specify the position of point/spot light or specify the direction (voctor) of directional light.

Constant Attenuation

Mathematically, intensity of light is divided by constant attenuation (I = 1/CA) where I is intensity and CA is constant attenuation. In this case, Intensity of the light will be totally unaffected by distance, continuing until it hits a surface, theoretically capable of illuminating an infinite area. This type of light can be seen from the sky (sunlight, moonlight and starlight), and other large and distant light sources.

Linear Attenuation

Mathematically, the decline of this type of light is linear, inversely proportional to the distance from the light source (I = 1/(LA* d) where I is intensity, LA is linear attenuation and d is distance). This means that the light intensity will diminish at a fixed rate as it travels from its source.

Quadratic Attenuation

Mathematically, the attenuation of a quadratic light is exponential (quadratic), expressed as "I = $1/(QA * d^2)$ ", meaning that the further the light travels from its source, the more it will be diminished. This creates a very sharp drop in light.

A directional is not subject to linear or quadratic attenuation since it is at an infinite distance.

Spot CutOff

Only available for spot light. This property specifies the light cone angle in degrees.

Spot Exponent

Only available for spot light. This property controls the amount of light falloff from the center of the spotlight

Spot Direction

Only available for spot light. This property specifies the direction (voctor) of

spot light.



Spot Light

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Engine O</u> → Sounds	bjects <u>← Previous page</u> <u>Next page</u> →
ODO: To change the footer's content go to Dr. Explain men	U Options : Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Sounds

- Sound Concepts
- Ambient Sound
- Static Sound
- Audacity

Made with help of Dr.Explain

)



Sound Concepts



Wave: Sound is made up of changes in air pressure in the form of waves.

Wavelength: The distance from the crest of one wave to the crest of the next.

Amplitude: The measure of the amount of energy in a sound wave.

Gain (Volume): A measure of how much a signal is amplified. Usually expressed in dB, higher gain values increase the amplitude of a signal, while lower gain values reduce it.

Frequency: The number of wave cycles per unit of time is called the frequency. For convenience, frequency is most often measured in cycles per second (cps) or the interchangeable Hertz (Hz) (60 cps = 60 Hz).



Pitch: For most practical purposes, the pitch of a sound can be said to be simply a measure of its frequency. Higher frequencies have higher pitch. High pitch means very rapid oscillation, and low pitch corresponds to slower oscillation. *Time stretching* is the process of changing the speed or duration of an audio signal without affecting its pitch. *Pitch scaling* or *pitch shifting* is the opposite: the process of changing the pitch without affecting the speed.

Stereo: Stereo sound divides sounds across two channels (recorded on two separate sources) then the recorded sounds are mixed so that some elements are channeled to the left and others to the right.

Mono: Commonly called mono sound, mono, or non-stereo sound, this early sound system used a single channel of audio for sound output.



Ambient Sound



Ambient Sound

Ambient sound is a background noise present in your scene. Ambient sound is very important in video games. It performs a number of functions including:

- Providing audio continuity in your scene.
- Preventing an unnatural silence when no other sound is present.
- Establishing or reinforcing the mood.

Ambient sounds in Vanda Engine 1 have no position or direction.

To access ambient sound editor, choose Create | Ambient Sound... from the main menu. Alternatively, you can click the ¹⁰⁰ button.

Name

Lets you choose a unique name for ambient sound.

Select Button

Lets you select the Vorbis file. Vorbis files (.ogg extension) compress to a smaller size than MP3 files, which reduces bandwidth and storage requirements.

Because ambient sound has no position or direction, you can import a stereo Vorbis file to simulate an illusion of directionality and audible

perspective.

Vanda Engine 1 works fine with 44100Hz Vorbis files.

Volume

Lets you change the volume of ambient sound. Accepted values are (0.0f, any]. Default value is 1.0.

Pitch

Lets you change the pitch of ambient sound. Accepted values are (0.0f, any]. Default value is 1.0.


Static Sound



Introduction to 3D sound in Vanda Engine 1

First of all, I should explain how 3D sound system works in Vanda Engine 1. There are 3 main things you need to be aware of: the listener and the sources, and attenuation by distance.

The Sources

Each sound source has a 3D coordinate which describes where it should be coming from. You should also assign an audio data to each sound.

The Listener

The use of a listener in 3D sound is important so that you have a reference point from which to hear sounds. The listener is the ears of the world, and as such, there's only one listener. Listener has a position and orientation. The position indicates where the listener currently is. The closer a listener is to a sound source's position, the louder the sound will get. The orientation is where the listener is facing. For example, if a listener is facing west and a sound is coming from the north, the sound will appear to come from the right. However if the user orientation changes and turns to face the north, the sound will now appear to come from straight ahead. Listener in Vanda Engine 1 is automatically attached to the current active camera and inherits its position and direction.

Attenuation by Distance

Attenuation is essentially the ability of a sound to lower in volume as the listener moves away from it. The effective attenuation of a source depends on many factors, among which distance attenuation and source and listener gain are only some of the contributing factors.

Vanda Engine 1 uses Inverse Distance Rolloff Model to attenuation the source gain by its distance from the listener position. The following formula describes the distance attenuation defined by the Inverse Distance Attenuation Model:

source gain = Reference Distance / (Reference Distance + Rolloff Factor *
(distance - Reference Distance));



Attenuation Inverse Distance Rolloff Model

Static Sound Editor

To access static sound editor, choose Create | Static Sound... from the main menu. Alternatively, you can click the button.

Name

Lets you choose a unique name for static sound.

Select Button

Lets you select the Vorbis file. Vorbis files (.ogg extension) compress to a smaller size than MP3 files, which reduces bandwidth and storage requirements.

stereo sounds aren't localized in Vanda Engine 1. If you use stereo data for your static sound, Vanda Engine 1 does not know about which channel to use in how much when panning the listener. Since Vanda Engine 1 is not sure what you want, it does nothing and doesn't even fade with distance. With mono, however, it does the calculations based on the attenuation formula, since it only has one channel to deal with.

Vanda Engine 1 works fine with 44100Hz Vorbis files.

Volume

Volume of static sound is specified by Attenuation by Distance and is irrelevant for static sound.

Pitch

Lets you change the pitch of static sound. Accepted values are (0.0f, any]. Default value is 1.0.

RollOff

Lets you specify the Roll Off factor.

Play

Lets you specify if Vanda Engine 1 should play the static sound data after loading it.

Loop

Lets you specify if Vanda Engine 1 should continually repeat the static sound data.

Reference Distance

Lets you specify the Reference Distance factor.

Maximum Distance

It's not used by current attenuation model and is reserved for future versions of Vanda Engine 1.





Audacity® is free, open source, cross-platform software for recording and editing sounds. It is available for Windows®, Mac®, GNU/Linux® and other operating systems. You can use Audacity to:

- Record live audio.
- Convert tapes and records into digital recordings or CDs.
- Edit WAV, AIFF, FLAC, MP2, MP3 or Ogg Vorbis sound files.
- Cut, copy, splice or mix sounds together.
- Change the speed or pitch of a recording.
- And more! See the complete list of features.

For more information, please refer to its official websie:

http://audacity.sourceforge.net/





Water Properties dialog lets you create the water object or edit its properties. To access Water Properties dialog, choose Create | Water... from the main

menu.



Water Properties Dialog

Name

Lets you specify a unique name for the water.

DuDv

Lets you assign a DuDv texture to the water. A DuDv map is similar to a normal map in that it stores directional information in a texture, except mostly just the Red and Green channels are used, so the bitmap has a red/yellow/green look. A DuDv map is actually a derivative of a Normal Map, and can easily be created with a variety of tools. It's usually best to blur the texture before converting it, to reduce the "jaggedness" of the distortion effect.



Default water DuDv used in Vanda Engine 1

Normal Map

Lets you assign a Normal Map to the water. A Normal Map is usually used to fake high-res geometry detail when it's mapped onto a low-res mesh. The pixels of the normal map each store a normal, a vector that describes the surface slope of the original high-res mesh at that point. The red, green, and blue channels of the normal map are used to control the direction of each pixel's normal.



Default water normal map used in Vanda Engine 1

By default, Vanda Engine 1 looks for these textures in Assets/Texture/Water folder. For more information, please see <u>Scene Options</u>.

Flow Speed

Lets you specify the flow speed of water.

UV

This options specifies how to project the DuDv and Normal Maps onto the surface of water. A value of 1 repeats each texture once, while a value of 3 repeats each texture three times on water surface.

Scale

Lets you specify the size of water. This value is multiplied by a 1*1 water.

Cx, Cy, Cz

Lets you specify the location of water.

Lx, Ly, Lz

Lets you specify the sun location for water.





Sky Properties dialog lets you create the Sky Dome object or edit its properties. To access Sky Dome Properties dialog, choose Create | Sky Dome... from the main menu.



Sky Dome Properties

Name

Lets you specify a unique name for the sky dome object.

Select...

Lets you assign a texture to the sky dome object.



By default, Vanda Engine 1 looks for this texture in Assets/Texture/Sky folder. For more information, please see <u>this section</u>.

Position X Y Z

Lets you specify the location (center) of sky dome object.

Radius

Lets you specify the radius of sky dome object.



Dampening

A value of 1 creates a Hemisphere sky dome. lower values convert it to an oval shape.



Slices

Lets you specify the slices of sky dome.



A slice of sky dome from the top view

Stacks

Lets you specify the stacks of sky dome.





Exponential

Does nothing. Reserved for future versions of Vanda Engine.

<u>Vanda Engine 1.3.3</u> \rightarrow <u>Vanda</u> \rightarrow Physics	\leftarrow Previous page <u>Next page</u> \rightarrow
TODO: To change the footer's content go to Dr Explain menu Ontions : I	

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Physics

- Introduction to PhysX
- Physics Concepts
- Physics Editor
- General PhysX Options
- PhysX XML format
- **Triggers**

Made with help of Dr.Explain

▶



Introduction to PhysX



Vanda Engine uses NVIDIA PhysX physics engine for real time physics interactions. PhysX is used by game engines such as Unreal Engine and Unity, and is used in many games such as Need for Speed: Shift, Mafia II, Batman: Arkham City, etc.

Built in Physics Editor of Vanda Engine supports the following PhysX elements:

- Static, Dynamic, and Kinematic Actors
- Box, Sphere, Capsule, Convex Hull, and Triangulated Mesh shapes
- Triggers
- Character Controller

Moreover, Vanda Engine imports the following PhysX elements from an external PhysX XML file:

- Scene properties
- Static and Dynamic Actors
- All types of PhysX shapes
- Materials



Physics Concepts

Polygon Mesh

In this section, a Polygon Mesh refers to *all* vertices, edges, and triangles of the *instanced geometry* imported from external 3D tools in COLLADA format.

Scenes and Actors

The most important physics objects are *scenes* and *actors*. A scene is a PhysX's representation of the world, and actors are individual elements of the world. To simulate physical world, Vanda Engine creates a PhysX scene and populates it with actors.

• Static Actor

static actor always stays at the same place and never moves around.

• Dynamic Actor

A dynamic actor will be affected by gravity and collide with other actors in the world. The dynamic actor can receive and apply forces and torque to make your instanced geometries move in a realistic way. You use dynamic actors for things that the player can push around. When you create a dynamic actor for an instanced geometry, the translation and rotation of instanced geometry is controlled by physics engine.

Kinematic Actor

A kinematic actor is not affected by forces, gravity, or collisions. It's driven explicitly by the translation and rotation of animated instanced geometry. Yet a kinematic actor can interact with dynamic actors.

Shapes

Physics shapes describe the spatial extent and collision properties of actors. In this section, 'shape' refers to physics shapes

Shape	Description	
Box	The Box shape is a basic cube-shaped primitive. It works great for doors, platforms, walls, etc.	
Sphere	The Sphere shape is a basic sphere- shaped primitive. It works great for balls, marbles, etc.	
Capsule	The Capsule shape is a basic capsule- shaped primitive.	
Convex Hull	A Convex Hull has no dents or indentations in it and contains all of the vertices of the polygon mesh.	
Triangulated Mesh	Triangulated Mesh shape takes the polygon mesh and builds its shape based on that mesh. It's far more accurate than using primitives for complicated instanced geometries.	

Triggers

Triggers are not affected by gravity and can't receive or apply forces and torque. Triggers have a unique set of three trigger messages (Enter, Stay, Leave) that are sent out when collision with a trigger occurs:

- Enter event is called when an actor enters the trigger.
- Stay event is called whilst an actor is inside the trigger.
- Leave event is called on leaving contact with the trigger.

Triggers in Vanda Engine are useful for triggering other actions such as activating the animation of an instanced geometry or loading another scene.

Vanda Engine supports static box-shaped triggers.

Character Controller

A character controller in Vanda Engine is used for third player controller. It's simply a capsule-shaped primitive which can be told to move in different directions. The character controller with then carry out the movement but be constrained by collisions. It will slide along walls, walk up the stairs, and push dynamic actors. See also <u>this section</u>.



Physics Editor



To access the Physics Editor, select your desired instanced geometry or skin and go to Modify | Geometry | Physics menu. Alternatively you can use button to access the Physics Editor. The Physics Editor lets you assign PhysX actors to skins and instanced geometries.

PhysX data is assigned to *instanced* geometries and skins.

Shapes

Lets you create the following shapes for actors:

- Box
- · Sphere
- · Capsule
- · Convex Hull
- Triangulated Mesh

Vanda Engine gives you several options to generate capsule or triangulated mesh shapes. If you intend to generate capsule or triangulated mesh shapes for your selected instanced geometry, you should try different methods to see which one works for you.

Percentage

This property is valid for triangulated mesh shapes and specifies the percentage

of number of polygon mesh vertices. Default values is 50, which means that triangulated mesh uses half the amount of polygon mesh vertices (For example if your polygon mesh contains 1000 vertices, triangulated mesh shape uses 500 vertices of the polygon mesh). Vanda Engine uses complex algorithms to collapse the polygon mesh until the triangulated mesh shape contains the vertex numbers specified by this property.

Density

Specifies the mass of actors. A value of 0 generates a static actor. Higher values generate dynamic actors.

Vanda Engine automatically assigns kinematic actors to animated instanced geometries and skins. Since density refers to static and dynamic actors, this property is meaningless for animated instanced geometries and skins.

Kinematic and dynamic actors can not use triangulated mesh shapes. Consequently:

1. Vanda Engine automatically sets density to 0 for triangulated mesh shapes (Meaning that they are static).

2. Vanda Engine does not let you specify triangulated mesh shapes for animated instanced geometries and skins (which use kinematic actors).

Table A and B summerize these concepts:

shape	possible actors for animated instanced	
	geometries and skins	
Box	Kinematic	
Sphere	Kinematic	
Capsule	Kinematic	

Convex	Kinematic	
Hull		

Table A: When you select an animated instanced geometry, Vanda Engine disables Density and Percentage fields and removes triangulated mesh shape from the Shape list.

shape	possible actors for non-animated instanced geometry	Density	Percentage
Box	Staitc/Dynamic	zero :static non- zero:	-
Sphere	Staitc/Dynamic	dynamic zero :static	_
		non- zero: dynamic	
Capsule	Staitc/Dynamic	zero :static non- zero: dynamic	-
Convex Hull	Staitc/Dynamic	zero :static	-

		non-	
		zero:	
		dynamic	
Triangulated	Staitc	-	1-100
Mesh			

Table B

Generate

Generates an actor for selected instanced geometry or skin:

1. For animated instanced geometries and skins, Vanda Engine generates kinematic actors.

2. For non-animated instanced geometries, Vanda Engine generates static or dynamic actors depending the value of Density:

- · If Density is set to 0, Vanda Engine generates a static actor
- \cdot If Density is set to a value higher than 0, Vanda Engine generates a dynamic actor.

Remove

Removes the actor from the selected instanced geometry or skin.



To access the General PhysX Options Editor, go to Modify | General PhysX Options menu.

General

Default Restitution

The amount of energy each actor retains after a collision. Typically this produces a bouncing effect. Default value is 0.3.

Default Static Friction

Static friction is the force that opposes a rigid body from changing its state from resting to motion. For instance, if you place a ball on an sloped plane, the static friction determines how easily the ball begins its initial slide or roll down the plane. Static friction has little or no effect after an object is moving. Default value is 0.5.

Default Dynamic Friction

Dynamic friction is how much a moving actor resists movement against another actor's surface. This is the force that tends to slow down an actor in motion. This force is generally proportional to any force that has been applied to the actor, so you can think of it as the ratio of frictional force to the applied force on the actor. Default value is 0.5.

Default Skin Width

Each actor has its own skin that goes around the actor at the width of the skin depth we are going to set. This skin is used for interpenetration between actors to account for inaccuracy when stacking actors on top of each other. The skin allows the actors to penetrate each other, at the width of the skin. The default skin width is set to 0.01.

Debug mode

Lets you enable or disable physics debug mode. In debug mode, you are able to see the Physics origin, character controller, triggers, actors, forces, etc.

Gravity

Apply gravity

Lets you activate or deactivate gravity. Gravity is applied to the character controller and dynamic actors.

Gravity X Y Z

Specifies the direction of gravity. By default, the gravity is applied in -Y direction.

Character Controller

Distance between main camera and character

Specifies the distance between the PhysX camera (attached to the character controller) and character controller. For more information about 3rd person camera in Vanda Engine, please visit <u>perspective viewport</u> section.

Force Power

Specifies the power of character force while pushing the dynamic actors.

Capsule Radius Specifies the radius of the character controller shape.

Capsule Height

Specifies the height of the character controller shape.

Speed

Specifies the movement speed of character controller.

Jumping

If checked and gravity is enabled, you can use Space key to jump the character controller.

Jump Power

Specifies the jump power of character controller. Higher values result in higher jumps.



PhysX XML format



Vanda Engine reads PhysX data exported from 3ds Max, Maya, or other 3D tools capable of exporting to PhysX XML format.

When you import a new PhysX XML file, Vanda Engine will remove current physics scene and actors, create a new scene based on the properties specified in XML file, generate a character controller, and populate the scene with the actors imported from PhysX XML file.

The following PhysX elements are imported:

- PhysX scene and its properties
- Static and dynamic actors
- All types of PhysX shapes
- Materials

Please be aware that built-in Physics Editor of Vanda Engine does not work with external PhysX scenes imported from 3D tools in XML format. Moreover, Vanda Engine does not know how to assign imported actors to instanced geometries and skins. Due to that, it's recommended to use built-in Physics Editor of Vanda Engine instead of importing the PhysX elements from other 3D tools.

To import a PhysX XML file, go to File | Import PhysX menu.

By default, Vanda Engine looks for Assets/PhysX/ path relative to current

installation folder to load PhysX XML files.



Triggers



Beside built-in Physics Editor to generate triggers and actors inside the editor, Vanda Engine can generate triggers from the 'Trigger Files'. Simply add a 'trigger' name somewhere in the COLLADA file name and import it to Vanda Engine. In this case, Vanda Engine automatically finds the bounding boxes of imported non-animated instanced geometries and builds the triggers based on the bounding boxes.

Vanda Engine does not render the instanced geometries or skins of trigger files.

Trigger files don't work with PhysX scenes exported from other 3D tools in XML format.

<u>Vanda Engine 1.3.3</u> \rightarrow <u>Vanda</u> \rightarrow Tools	\leftarrow Previous page <u>Next page</u> \rightarrow
IODO: To change the footer's content go to Dr Explain menu Ontions • I	· · · · · · · · · · · · · · · · · · ·

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of Tools

- Camera and Rendering
- Water Attachment
- General Options
- Scene Options

Made with help of Dr.Explain

)



Camera and Rendering

To access the Camera dialog, choose Tools | Camera... from the main menu.



Camera Dialog

This dialog lets you activate the default Free or 3rd Person camera as well as external cameras imported from 3D tools in COLLADA format. Only one camera can be active at a time.

We are planning to support a feature that lets you capture a AVI video from the perspective of animated external camera. If selected external camera does not contain animation, Vanda Engine will render the scene to a TGA file.



Water Attachment



To access the Water Attachment dialog, select a skin o instanced geometry in the VScene and choose Tools | Water Attachment... from the main menu.



Water Attachment Dialog

To attach a skin or instanced geometry to a specified water, choose the water name from the 'Water Names' list and click attach. When you attach a skin or instanced geometry to the selected water, Vanda Engine 1 will render the reflection of the attached skin or instanced geometry on the surface of specified water.

Water attachment does not work with material based selection.



General Options



To access General Options, choose Tools | General Options... from the main menu. This dialog lets you change several options of Vanda Engine 1.

Some of these changes will take affect the next time you run Vanda Engine 1 editor.

Shader

Lets you enable or disable shaders. If Vanda Engine 1 can initialize GLSL or shaders, this checkbox becomes enabled by default. Otherwise, this checkbox becomes grayed and you can not change its status.

Multisampling

Multisampling is a type of anti-aliasing, a technique used in computer graphics to improve image quality. This entry allows you to enable or disable this feature and specify number of samples. If your OpenGL implementation supports multisampling, this feature is enabled by default. Otherwise, this checkbox becomes grayed and you can not change its status. More samples increase the quality, but affect the performance.

Anisotropic Texture Filtering

Anisotropic filtering is a method of enhancing the image quality of textures on surfaces of geometries that are at oblique viewing angles with respect to the active camera. This entry allows you to enable or disable this feature and specify number of samples. If your OpenGL implementation supports anisotropic texture filtering, this feature is enabled by default. Otherwise, this checkbox becomes grayed and you can not change its status. More samples increase the quality, but affect the performance.

Framebuffer Object

Framebuffer Objects are a mechanism for rendering to images other than the default OpenGL Default Framebuffer. They are OpenGL Objects that allow you to render directly to textures, as well as blitting from one framebuffer to another. This entry allows you to enable or disable this feature. If your implementation does not support Framebuffer Objects, this checkbox becomes grayed and Vanda Engine 1 uses old rendering styles to render your scene.

Vertex Buffer Object

A Vertex Buffer Object is an OpenGL feature that provides methods for uploading data (vertex, normal vector, color, etc.) to the video device for nonimmediate-mode rendering. VBOs offer substantial performance gains over immediate mode rendering primarily because the data resides in the video device memory rather than the system memory and so it can be rendered directly by the video device. If your OpenGL implementation does not support this feature, this checkbox becomes grayed and Vanda Engine 1 uses immediate mode rendering. Otherwise, this checkbox lets you enable or disable Vertex Buffer Objects.

Disable VSync

VSync stands for Vertical Synchronization and synchronizes your FPS with your monitor's refresh rate. The purpose is to eliminate something called "tearing". Every CRT monitor has a refresh rate. It's specified in Hz (Hertz, cycles per second). It is the number of times the monitor updates the display per second. FPS is how many frames the video card can draw per second. Higher is obviously better. However, during a fast paced game, your FPS rarely

stays the same all the time. It moves around as the complexity of the image the video card has to draw changes based on what you are seeing. This is where tearing comes in.

Tearing is a phenomenon that gives a disjointed image. The idea is as if you took a photograph of something, then rotated your vew maybe just 1 degree to the left and took a photograph of that, then cut the two pictures in half and taped the top half of one to the bottom half of the other. The images would be similar but there would be a notable difference in the top half from the bottom half. This is what is called tearing on a visual display. It doesn't always have to be cut right in the middle. It can be near the top or the bottom and the separation point can actually move up or down the screen, or seem to jump back and forth between two points.

VSync solves this problem by creating a rule that says the back buffer can't copy to the frame buffer until right after the monitor refreshes. If you're playing a game that has a framerate that routinely stays above your refresh rate, then VSync will generally be a good thing. However if it's a game that moves above and below the refresh rate or even worse, lays at an FPS that is just below the refresh rate, then VSync can become annoying. In this case, it's a better idea to disable VSync.

This options lets you enable or disable VSync.

Water Reflection

Lets you enable or disable whole scene water reflections.

Show Welcome Dialog

Lets you show or hide the Welcome Dialog while initializing Vanda Engine 1.


Scene Options



To access Scene Options, choose Tools | Scene Options... from the main menu. This dialog lets you change the default naming conventions of your current VScene. Vanda Engine 1 uses these naming conventions to load the extra textures.

Naming Conventions

While importing the skins and instanced geometries, Vanda Engine 1 looks for the following naming conventions to load extra textures located beside the diffuse texture:

- Normal Map: Diffuse Texture Name + "_n"
- Alpha Map: Diffuse Texture Name + "_a"
- · Gloss Map: Diffuse Texture Name + "_g"
- Shadow Map: Diffuse Texture Name + "_s"
- Dirt Map: Diffuse Texture Name + "_d"

For example, if Vanda Engine imports a diffuse texture called "diffuse.dds" in "C:/Texture/" path for a geometry, by default it looks for the following texture names in "C:/Texture/" to load extra textures:

- Normal Map: diffuse_n.dds
- · Alpha Map: diffuse_a.dds
- · Gloss Map: diffuse_g.dds

- Shadow Map: diffuse_s.dds
- Dirt Map: diffuse_d.dds

Banner

This feature lets you specify the loading banner of current VScene.

Made with help of Dr.Explain

← Previous page <u>Next page</u> →
Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Scripting

- Introduction to Lua
- **Functions**

Made with help of Dr.Explain

►]



Introduction to Lua

()

What's Lua

To quote the official definition, Lua is a powerful, fast, lightweight, embeddable scripting language.

Installation

Please download and install the Lua package from Windows from <u>Google</u> <u>Code</u>.

IDE

It's always a good idea to have an IDE on hand. The IDE that comes with the Lua package is called SciTE.

Running SciTE

Please click the SciTE shortcut in order to run the IDE.

Saving the Script

After writing your script, go to File | Save As menu, give your file a name, and make sure to include .lua at the end of your file name. Otherwise, Vanda Engine won't recognize your lua script.

Made with help of Dr.Explain



Functions



Animating 3D characters and animated instanced geometries

Vanda Engine provides all mechanisms needed to load, render and animate 3D models. Loading the 3D models is done with the interface, and rendering the models is done by Vanda Engine automatically. Vanda Engine lets you manipulate the animations of the scenes via the script.

There are two different approaches for animation:

- Looping animation (handled by BlendCycle and ClearCycle)
- One time animation (handled by ExecuteAction and ReverseExecuteAction)

As the names suggested it is possible to both repeat the animation as long as needed, and to play the animation just once.

All of the functions that manipulate animations deal with animation clips. Animation clips can be used to separate different pieces of a set of animations. For example, an animation might have a character walk, then run. The walking and running animations can be separated as two different clips. Clips can also be used to separate the animations of different characters in the same scene, or even different parts of the same character (such as upper and lower body). If you don'e specify animation clips for your animated geometries, Vanda

Engine creates a default animation clip called defaultClip.

Animation is a per scene feature. It means that when you activate an animation clip of a scene, all of the geometries affected by that animation clip are animated and all of the nodes and hierarchies (including static nodes of static geometries) are recalculated as well. Due to that, it's strongly recommended that you export the static geometries separately.

BlendCycle

bool BlendCycle (char* sceneName, char* clipName, float weight, float delay)

Interpolates the weight of an animation cycle.

This function interpolates the weight of an animation cycle to a new value in a given amount of time. If the specified animation cycle is not active yet, it is activated.

Parameters

- \cdot sceneName: The external scene name that its animation clip should be blended.
- clipName: The clip name of the animation cycle that should be blended.
 If your scene does not contain animation clip, set this parameter to "defaultClip".
- weight: The weight to interpolate the animation cycle to.
- · delay:The time in seconds until the new weight should be reached.

Returns

One of the following values:

- true if successful
- false if an error happened

example

BlendCycle("hand amb.dae", "byebye", 1.0, 0.5)

ClearCycle

bool ClearCycle (char* sceneName, char* clipName, float delay)

Fades an animation cycle out.

This function fades an animation cycle out in a given amount of time.

Parameters

 \cdot sceneName: The external scene name that its animation clip should be cleared.

• clipName: The clip name of the animation cycle that should be cleared. If your scene does not contain animation clip, set this parameter to "defaultClip".

 \cdot delay: The time in seconds until the the animation cycle is completely removed.

Returns

One of the following values:

- true if successful
- false if an error happened

example

ClearCycle("hand amb.dae", "byebye", 0.5);

ExecuteAction

bool ExecuteAction(char* sceneName, char* clipName, float delayIn, float delayOut, float weightTarget = 1.0f, bool lockToLastKeyframe = false)

Executes an animation action.

This function executes an animation action.

Parameters

sceneName: The external scene name that its animation clip should be

executed.

clipName: The clip name of the animation cycle that should be executed.
 If your scene does not contain animation clip, set this parameter to "defaultClip".

 \cdot delayIn: The time in seconds until the animation action reaches the full weight (specified by weightTarget) from the beginning of its execution.

 \cdot delayOut: The time in seconds in which the animation action reaches zero weight at the end of its execution.

• lockToLastKeyframe: Lets you lock the clip name to its last keyframe at the end of its execution. For example, if you want to open a door, you should tell Vanda Engine to lock the animation to the last keyframe at the end of its execution. Otherwise, the door returns back to it status specified by first keyframe.

Returns

One of the following values:

- true if successful
- false if an error happened

example

ExecuteAction("SampleInstancedGeometries.dae", "defaultclip", 0.5, 0.5, 1, true)

ReverseExecuteAction

bool ReverseExecuteAction(char* sceneName, char* clipName)

Executes an animation action in reverse order. If the specified action is not in 'action' status this function returns false. If the specified action is in action status, but its current keyframe is 0, this function returns false as well.

Parameters

sceneName: The external scene name that its animation clip should be

executed.

clipName: The clip name of the animation cycle that should be executed.
 If your scene does not contain animation clip, set this parameter to "defaultClip".

Returns

One of the following values:

- true if successful
- false if an error happened

example

ReverseExecuteAction("SampleInstancedGeometries.dae", "defaultclip")

Loading another VScene

It's possible to load another VScene while entering or leaving a trigger. Simply use the LoadScene function and assign the script to the trigger event.

LoadVScene

bool LoadVScene(char* sceneName)

Loads the VScene specified by sceneName argument.

Parameters

sceneName: The VScene name that should be loaded.

Returns

One of the following values:

- true if successful
- false if an error happened

example LoadVScene("Second Scene") This opens a saved VScene name called Second Scene.vin.

Sounds

PlaySound

int PlaySound(char* SoundName1, char* soundName2, ..., char* loopStatus)

Plays the sounds specified by SoundName arguments.

Parameters

SoundName[1...n]: The sound names to be played. loopStatus: Optional. Valid string is "VANDA_LOOP". If "VANDA_LOOP" is specified, VE loops the sounds, otherwise it plays the specified sounds once.

example

PlaySound("static1", "static2", "VANDA_LOOP")

StopSound

int StopSound(char* SoundName1, char* soundName2, ...)

Stops the sounds specified by SoundName arguments.

Parameters

SoundName[1...n]: The sound names to be stopped.

example

```
StopSound( "static1", "static2" )
```

PauseSound

int PauseSound(char* SoundName1, char* soundName2, ...)

Pauses the sounds specified by SoundName arguments.

Parameters

SoundName[1...n]: The sound names to be paused.

example

PauseSound("static1", "static2")

Made with help of Dr.Explain

Vanda Engine 1.3.3 \rightarrow Vanda \rightarrow Publishing	\leftarrow Previous page <u>Next page</u> \rightarrow
TODO: To change the footer's content go to Dr.Explain menu Options : Proje HTML Template and Layout	ct Settings : HTML (CHM) Export : Setup

The contents of Publishing

Properties

Made with help of Dr.Explain

Þ



HTML Template and Layout



To access publish editor, choose File | Publish... from the main menu.

VandaEngine RTI Pro Version 1.2.0		- Ø X
File View Create Modify PhysX Tools Wind		
📄 📹 🔞 🦊 県 🖏 🏠 🔊 🤇	S 🕄 🕕 🔁 🔘 🛞	🤕 🕒 😳 🥪
📷 front	Publish Project	External Scenes
	Project Name My Project	
	C:\ Destination	×
•••	E:\New Engine\VandaEngine\win32\Zib Icon	External Elements
	E:\New Engine\VandaEngine\win32\Zib Splash	
	Available Scenes Scenes To Be Published	
Perspective	3.vin	Engine Objects
	Mark current selection as first scene	2 🗴
		PhysX Elements
	If current scene is in publish list, save it automatically Publish Cancel	
		×
Ln 73> Scene Saved Successfully	In 1> Errors and warnings related to the COLLADA files	

Project Name

Lets you name your project

Destination

Opens a standard dialog that lets you select the destination directory.

Vanda Engine publishes all of the files in 'Destination/Project Name/' directory. Make sure that this directory is empty.

Icon

Opens a standard dialog that lets you select the icon of executable.

Splash

Opens a standard dialog that lets you select the splash image.

Dimensions of splash image should be 550 * 314.

Available Scenes

Lists all of the available VScenes.

Scenes to be Published

Lists all of the VScenes to be published.

Use arrow buttons to add/Remove the VScenes to be published.

Mark Current Selection as First Scene

Lets you mark the current selection in the 'Scenes to be Published' list as the first scene to be loaded.

If current scene is in publish list, save it automatically

If current scene is in publish list, Vanda Engine uses save command to save the current scene.

<u>Vanda Engine 1.3.3</u> \rightarrow <u>Vanda</u> \rightarrow Exporting	\leftarrow Previous page <u>Next page</u> \rightarrow
ODO: To change the footer's content go to Dr. Explain menu Options :	Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Exporting

OpenCOLLADA

PhysX

Made with help of Dr.Explain

►]

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Exporting</u> → OpenCOLLADA	\leftarrow Previous page Next page \rightarrow
ODO: To change the footer's content go to Dr.Explain menu Options	: Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setu** HTML Template and Layout

The contents of OpenCOLLADA

- Exporting from 3ds Max
- Exporting from Maya

Made with help of Dr.Explain

•



Exporting from 3ds Max using OpenCOLLADA



• Download and install the OpenCOLLADA plug in for 3ds Max from https://github.com/KhronosGroup/OpenCOLLADA/wiki/OpenCOLLADA-Tools

- Open 3ds Max or restart if it was open when you installed.
- · Open up your scene, choose File | Export
- Select OpenCOLLADA (*.dae) from the drop down box, give the file a name, and press Save button.
- This will open the OpenCOLLADA Export dialog.

There are three main areas in OpenCOLLADA Export dialog: Standard Options, Geometry, and Animation.

Standard Options

Bake Matrices

The transforms of each scene node and their animations will be exported as matrices.

Relative Paths

Whether the absolute or relative file path will be written to the COLLADA document. Mostly impacts texture and shader filenames.

Copy Images

This will copy all the texture files (diffuse, normal, alpha, etc) into "\images" folder above your .DAE file.

Geometry

Normals

Lets you export the normals of meshes.

Triangulate

Divides your polygons into triangles

Tangents/Binormals

Exports the geometry tangents and binormals for all given map channels of meshes, as well as their indices within the mesh tessellation.

Animation

Enable export

Lets you export animations.

Sample animation

Specifies the start-frame and end-frame of the animation. It is by default set to the timeline range of your scene.

• To export your scene to be loaded in Vanda Engine 1, you must uncheck *Bake Matrices*. Since Vanda Engine 1 automatically triangulates the meshes, *Triangulate* is irrelevant. *Relative Paths* and *Copy Images* are irrelevant, since Vanda Engine 1 uses its own parser to read the file names locally. So uncheck these two options. If your model contains textures, you should copy its textures in appropriate places. In this case, copy diffuse maps to data/textures folder and other maps such as normal and alpha maps to data/extra folder. Other options such as normals, tangents, binormals, and animations are optional and depend on your purposes.

Press OK to export your scene.

Made with help of Dr.Explain

•



Exporting from Maya using OpenCOLLADA



• Download and install the OpenCOLLADA plug in for Maya from https://github.com/KhronosGroup/OpenCOLLADA/wiki/OpenCOLLADA-Tools

• Open Maya or restart if it was open when you installed.

 \cdot Go to windows | Preferences | Plug-in Manager and make sure the OpenCOLLADA plug in is checked and in use.

- Open up your scene and choose File | Export All.
- · Click Options button. This will open the Export All Option dialog.
- Select OpenCOLLADA from the drop down box.

Export All Options			
Edit Help			
	Default file extensions Freserve references		•
File Type Specific Option	ns		-
	Bake transforms Bake transforms Relative paths Triangulate Cgfx file references	Copy textures	
	Sampling Curve constrain	Static curve removal	
Filter export	Export: Polygon meshes Animations Animations Invisible nodes Texture coordinates Normals Vertex colors Texture tangents Referenced materials only	 Lights Joints and skin Default cameras Normals per vertex Vertex colors per vertex Geometric tangents Materials only 	II
▼ XRef options	Export references	V Dereference	
	Export camera as lookat XFov If neither XFov and YFov are selected. Export double precision ouble precision is selected, the maximur digits to be printed is increased from 7 to	m number of significant	
Export All	Apply	Close	-

A picture showing the available COLLADA exporter options

General Export Options

Bake Matrices

The transforms of each scene node and their animations will be exported as

matrices.

Relative Paths

Whether the absolute or relative file path will be written to the COLLADA document. Mostly impacts texture and shader filenames.

Copy Images

This will copy all the texture files (diffuse, normal, alpha, etc) beside your .DAE file.

Triangulate

Divides your polygons into triangles

Cgfx file references

CgFX format is a shader format used by Cg to define the appearance, or effects, of 3D objects. This file stores source code in a plain text format and is referenced by various games and other programs that use NVIDIA's CgFX technology.

This option will copy all the Cgfx files beside your .DAE file.

Sampling

makes your animation export correctly and smooth.

Curve Constrain

Constrains the sampled times to a curve's range.

Static Curve Removal

A curve is considered to be static if it would return the same value regardless of the evaluation time. This basically means that the values of all the keys are the same and the y component of all the tangents is 0. This option removes static curves which influence nothing.

Filter Export

Polygon Meshes

A polygon mesh is a collection of vertices, edges and faces that defines the shape of a polyhedral object in 3D computer graphics and solid modeling. This option lets you export objects created with polygon meshes.

Lights

Lets you export ambient, directional, point and spot lights.

Cameras

Exports perspective and orthographic cameras.

Joints and Skin

If you have characters with skeleton in your scene, you should check this option. This option exports all the information required for skinning algorithm.

Animations

Self explanatory. Lets you export animations.

Invisible nodes

An invisible node is the node of any hidden object in your scene. This option lets you export invisible nodes.

Default cameras

Exports default Maya cameras. To export default Maya cameras, you should also check invisible nodes option to activate this option.

Texture coordinates

This option exports all texture coordinates.

Normals

Lets you export normal vectors.

Normals per vertex

COLLADA files can use one normal per polygon per vertex, especially to account for smoothing related to the same vertex being part of multiple polygons. Thus it is possible to have far more normals than vertices and in the combined index, each vertex can be associated with multiple normals. This option lets you export normals per polygon per vertex.

Vertex colors

Each vertex in a mesh can optionally store a RGB color value, called vertex color. These can be used for a variety of interesting effects and shader inputs. Vertices can also store vertex alpha, which can be used for multitexturing, transparency, or ambient occlusion.

This option lets you export vertex colors.

Vertex colors per vertex

A single vertex can be assigned multiple vertex colors. Thus it is possible to have far more colors than vertices and in the combined index, each vertex can be associated with multiple colors. This option lets you export multiple colors per vertex.

Texture tangents

Exports tangents and binormals in texture space. This is the typical use-case for "tangent space" normal maps.

Texture tangents

Exports tangents and binormals in geometric space. This is the typical use-case for "object space" or "world space" normal maps.

Reference materials only

Exports the materials used in your scene. Note that this option does not disable other export options.

Materials only

Disables other export options and exports materials only.

XRef options

Export references

Exports external references used in your scene.

Dereference

References external elements in COLLADA file. Here's an example of de-referencing in COLLADA file:

<instance_geometry url="file:///models/car.dae#carGeometry" />

This <instance_geometry> references the <geometry> element whose id is carGeometry in the document file:///models/car.dae.

Camera options

Export camera as lookat

Describes the camera with three parameters:

- · Camera position.
- Interest position.
- The direction that points up.

Here's an example of a camera exported as lookat:

<lookat>0 2 0 2.5 2 -4 0 1 0 </lookat>

It simply explains that the camera is at (0, 2, 0) position, looks at (2.5, 2, -4) position and its up vector is (0, 1, 0).

XFov

angle of view describes the angular extent of a given scene that is imaged by a

camera.

Field of view may be measured horizontally (from the left to right edge of the frame), vertically (from the top to bottom of the frame), or diagonally (from one corner of the frame to its opposite corner).

This option lets you export horizontal field of view, in degrees.

YFov

angle of view describes the angular extent of a given scene that is imaged by a camera.

Field of view may be measured horizontally (from the left to right edge of the frame), vertically (from the top to bottom of the frame), or diagonally (from one corner of the frame to its opposite corner).

This option lets you export vertical field of view, in degrees.



This image depicts horizontal and vertical FOVs.

Precision options

Export double precision

If this option is selected, the maximum number of significant digits to be

printed is increased from 7 to 17 significant digits.

• These are settings for Maya when exporting to a DAE file for Instancing COLLADA in Vanda Engine 1:

Option	Recommended	Issues
	setting	
Bake transforms	No	If you check this option, the Vanda Engine 1 crashes.
Relative paths	Irrelevant	Vanda Engine 1 only reads file names in COLLADA files and does not read COLLADA paths at all.
Copy textures	Irrelevant	You should manually copy diffuse maps to data/textures folder and other maps such as normal and alpha map to data/extra folder. Note that Vanda Engine 1 reads the files locally.
Triangulate	Irrelevant	Vanda Engine 1 automatically triangulates the meshes.
Cgfx file	No	
reference		
Sampling	No	
Curve constrain	No	
Static curve removal	No	

Polygon meshes	Yes	
Lights	Yes	Vanda Engine 1 has a built in light editor. COLLADA lights are not as accurate as Vanda Engine 1 lights. However Vanda Engine 1 is able to load COLLADA lights as well.
Cameras	Yes	Vanda Engine 1 has a default 3rd person camera. COLLADA cameras in Vanda Engine 1 are used for cut-scenes.
Joints and skin	Yes	
Animations	Yes	
Invisible Nodes	Yes	
Default cameras	Yes	
Texture coordinates	Yes	
Normals	Yes	
Normals per vertex	?	
Vertex colors	?	
Vertex	?	

colors per vertex		
Texture tangents	Yes	Normal and parallax maps in Vanda Engine 1 are based on texture tangents.
Geometric tangents	No	
Referenced materials only	Yes	Vanda Engine 1 only reads referenced materials.
Materials only	No	Vanda Engine 1 only reads referenced materials.
Export references	No	
Dereference	No	
Export camea as look at	No	If you enable this option, Vanda Engine 1 positions the cameras incorrectly.
XFov	No	
YFov	Yes	
Export double precision	No	

• You should also consider the following facts for Instancing COLLADA in Vanda Engine 1:

1. You should export each character with its own animations as a separate COLLADA scene.

2. Vanda Engine 1 uses a different algorithm to render and animate grasses. You should say Vanda Engine 1 that your exported file refers to grasses. To do this, add a "grass" prefix to your COLLADA file name. Such file should only consist of grass geometries. This section explains how to create, export, and import grasses.

3. Vanda Engine 1 uses a simple algorithm to create PhysX triggers based on the information of a COLLADA file. You should say Vanda Engine 1 that your exported file refers to triggers. To do this, add a "trigger" prefix to your COLLADA file name. Such file should only consist of trigger geometries. This section explains how to create, export, and import triggers.

 \cdot Click Export All, give your file a name based on the above facts, and click Export.

Made with help of Dr.Explain

<u>Vanda Engine 1.3.3</u> → <u>Vanda</u> → <u>Exporting</u> → PhysX	← Previous page Next page →
ODO: To change the footer's content go to Dr. Explain menu Options	s : Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup HTML Template and Layout**

The contents of PhysX

- PhysX plugin for 3ds Max
- PhysX plugin for Maya

Made with help of Dr.Explain

•



Exporting PhysX from 3ds

PhysX plug-in for 3ds Max is a plug-in that adds certain PhysX SDK functionality, like rigid body simulation, to Autodesk 3ds Max and allows physics calculations to be performed directly in viewports.

You can download PhysX plugin for 3ds Max from

https://developer.nvidia.com/rdp/physx-downloads (login is required).



Vanda Engine 1 imports PhysX data in XML format.

Made with help of Dr.Explain



PhysX Plugin for Maya

PhysX plug-in for Maya adds PhysX SDK simulation functionality to Autodesk Maya.

You can download PhysX plugin for Maya from

https://developer.nvidia.com/rdp/physx-downloads (login is required).



Vanda Engine 1 imports PhysX data in XML format.

Made with help of Dr.Explain

<u>Vanda Engine 1.3.3</u> \rightarrow <u>Vanda</u> \rightarrow Reference	\leftarrow Previous page Next page \rightarrow
	· · · · · · · · · · · · · · · · · · ·
ODO: To change the footer's content go to Dr. Explain menu Options : P	Project Settings : HTML (CHM) Export : Setup

TODO: To change the footer's content go to Dr.Explain menu **Options : Project Settings : HTML (CHM) Export : Setup** HTML Template and Layout

The contents of Reference

Reference

Made with help of Dr.Explain

 \rightarrow



Reference



Note: This reference is not in APA format. APA format will be available soon. en.wikipedia.org/wiki/Multisample_anti-aliasing http://en.wikipedia.org/wiki/Vertex_Buffer_Object http://www.opengl.org/wiki/Framebuffer_Object developer.download.nvidia.com/.../cascaded_shadow_maps http://en.wikipedia.org/wiki/Bounding_volume en.wikipedia.org/wiki/Octree http://udn.epicgames.com/Three/ShadowingReference.html http://en.wikipedia.org/wiki/Back-face_culling http://wiki.polycount.com/NormalMap/ http://wiki.polycount.com/SpecularGlossMap http://wiki.splashdamage.com/index.php/Specular_Maps http://support.nextlimit.com/display/maxwelldocs/Alpha+channel en.wikipedia.org/wiki/Alpha_mapping http://www.monstrmoose.com/MMrepository/Halo_Tools/2_Vista/Tutorial/Halc http://wiki.splashdamage.com/index.php/Diffuse_Maps https://ourbricks.zendesk.com/entries/407171-using-opencollada-with-maya https://ourbricks.zendesk.com/attachments/token/t6ptzm65giafv8c/? name=Exporting+from+3dStudio+Max+using+OpenCollada.pdf http://opencollada.org/OpenCOLLADA%20techniques.pdf http://www.terathon.com/wiki/index.php/Autodesk_3ds_Max http://www.fileinfo.com/extension/cgfx

http://www.frictionalgames.com/forum/thread-18392.html http://sourceforge.net/project/shownotes.php?release_id=366704 tp://en.wikipedia.org/wiki/Texture_mapping http://collada.org/public_forum/showthread.php/1223-Deindexing-Unifying-Vertex-Normal-Texccord-Indices http://wiki.polycount.net/VertexColor http://collada.org/public_forum/showthread.php/1307-textangent-vs-tangent http://pycollada.github.com/reference/generated/collada.camera.PerspectiveCam http://collada.org/mediawiki/index.php/Using_URIs_in_COLLADA http://scenejs.wikispaces.com/COLLADA+Export+from+Maya http://en.wikipedia.org/wiki/Angle of view http://www.cse.unr.edu/~quiroz/inc/TextureNotes.pdf http://download.novedge.com/Brands/Alias/Helps/Maya6.5/en_US/Live/otherm http://3d.about.com/od/Creating-3D-The-CG-Pipeline/ss/Maya-Training-Series-Chapter-1-Learing-The-User-Interface_3.htm http://www.3dmax-tutorials.com/General_Viewport_Concepts.html http://www.expertrating.com/courseware/3DCourse/3DMax-UserInterface-2.asp http://www.opensg.org/htdocs/doc-1.8/Windows.html http://help.apple.com/motion/mac/5.0/en/motion/usermanual/index.html#chapte http://help.apple.com/motion/mac/5.0/en/motion/usermanual/index.html#chapte UserManual-90919TDC-SW5 http://wiki.blender.org/index.php/Doc:2.4/Manual/3D_interaction/Navigating/Ca http://www.matrix44.net/cms/notes/opengl-3d-graphics/coordinate-systems-inopengl http://findicons.com/icon/74226/pencil?id=406996 http://www.freesfx.co.uk (for ambient sound) http://www.opengl.org/sdk/docs/man/xhtml/glPolygonMode.xml http://docs.unity3d.com/Documentation/Components/class-Light.html http://budgetdecorating.about.com/od/homedecoratingtermsad/g/Ambientlighting.htm

http://ogldev.atspace.co.uk/www/tutorial18/tutorial18.html http://en.wikipedia.org/wiki/Specular_highlight https://developer.valvesoftware.com/wiki/Constant-Linear-Quadratic_Falloff http://www.scenebeta.com/tutorial/lecciones-19-22-y-practicas-05-08 http://www.mediacollege.com/audio/ambient/ http://whatis.techtarget.com/definition/Ogg-Vorbis http://www.indiana.edu/~emusic/acoustics/frequency.htm http://www.fi.edu/fellows/fellow2/apr99/soundvib.html http://oceanlink.island.net/oinfo/acoustics/soundwave.gif http://manual.audacityteam.org/ http://en.wikipedia.org/wiki/Pitch_shift http://en.wikipedia.org/wiki/Audio_time-scale/pitch_modification http://www.webopedia.com/TERM/M/monophonic_sound.html http://www.gamedev.net/page/resources/_/technical/gameprogramming/programming-3d-sound-with-openal-in-windows-r1958-r2008 http://www.gamedev.net/page/resources/_/technical/game-programming/aguide-to-starting-with-openal-r2008 http://www.idevgames.com/forums/thread-4992.html http://forum.qbasicnews.com/index.php?topic=8386.0;wap2 OpenAL 1.1 specification http://udn.epicgames.com/Three/DistanceModelAttenution.html http://en.wikipedia.org/wiki/Anisotropic_filtering http://hardforum.com/showthread.php?t=928593 http://wiki.polycount.com/DuDvMap http://i.stack.imgur.com/uJVXS.png http://wiki.polycount.com/NormalMap http://tex.stackexchange.com/questions/22894/some-extensions-to-pstsolides3d http://support.quest3d.com/index.php?title=Chapter_2.9:_Landscapes http://www.ozone3d.net/tutorials/glsl_fog/p02.php http://physxinfo.com/wiki/PhysX_plug-in_for_Maya

http://physxinfo.com/wiki/PhysX_plug-in_for_3ds_Max http://www.amateurphotographer.co.uk/how-to/tech-explained/536453/depthof-field-explained http://www.cambridgeincolour.com/tutorials/depth-of-field.htm http://answers.yahoo.com/question/index?qid=20101130063926AAVNXnQ http://en.wikipedia.org/wiki/Geometry_instancing http://en.wikipedia.org/wiki/Polygon_mesh https://developer.nvidia.com/sites/default/files/akamai/physx/Docs/ScenesAndA https://developer.nvidia.com/sites/default/files/akamai/physx/Docs/Shapes.html http://docs.unity3d.com/Documentation/Components/class-CharacterController.html http://docs.unity3d.com/Documentation/Manual/Physics.html http://www.conitec.net/english/gstudio/faq.php http://en.wikipedia.org/wiki/Octree http://www.lighthouse3d.com/tutorials/view-frustum-culling/ http://www.flipcode.com/archives/Introduction_To_Octrees.shtml http://developer.bada.com/badaforum/board/thread/view.do? boardName=tools&messageId=46026 http://cal3d.sourceforge.net/docs/api/html/classCalMixer.html#a6 Unity 3.x Game Development Essentials http://www.orbaz.com/documentation/particleflowtools/box2/reference/06Phys2 http://softimage.wiki.softimage.com/index.php? title=Advanced_PhysX_Parameters

 $|\mathbf{F}|$

Made with help of Dr.Explain

4



FAQ

E

Can I use Vanda Engine for commercial use?

Yes, Vanda Engine is free for commercial use.

What operating systems does Vanda Engine support?

Vanda Engine is available for Windows XP, Windows Vista, Windows 7 and Windows 8.

Is my 3D tool compatible with Vanda Engine?

Vanda Engine is compatible with any 3D tool capable of exporting to COLLADA format. We have tested and verified that Vanda Engine supports 3ds Max, Maya, Blender, Cinema 4D, Sketchup and AC3D.

How can I export my scenes from my 3D tool to COLLADA format?

First of all, make sure that your 3D tool supports COLLADA format. Secondly, read the reference manual of your 3D tool regarding the COLLADA exporter. Finally, read the Vanda Engine tutorials about exporting the scenes from different 3D tools to COLLADA format.

Why Vanda Engine does not import my exported scenes in COLLADA format?

Make sure you export the scene in COLLADA 1.4 format.

What texture types does Vanda Engine support?

Vanda Engine reads DXT1, DXT3 and DXT5 compression types saved as a

DDS file. For more information, please refer to <u>this</u> article.

Does Vanda Engine support alpha channel of DDS files?

Vanda Engine assumes that DXT1 compression has no alpha channel and DXT3 or DXT5 compressions have an alpha channel. If your geometry shouldn't have an alpha channel, use DXT1 compression type.

Why Vanda Engine does not read textures?

Make sure your textures exist in the directory specified in the COLLADA file. Moreover, the dimensions of your DDS textures should be power of 2.

Why Vanda Engine does not read normal, alpha, gloss, shadow and dirt maps?

1. Vanda Engine uses simple naming conventions to load extra textures. For more information regarding the naming conventions, please refer to <u>this</u> section.

2. Make sure that your maps are beside the diffuse texture.

3. Texture dimensions should be power of 2.

How to assign parallax maps to instanced geometries and skins?

In order to assign a parallax map, your normal map should have an alpha channel related to parallax mapping. Moreover, you can change the properties of parallax maps via the Skin and Instance Geometry dialog.

Does Vanda Engine import different elements of a COLLADA file?

Vanda Engine interprets each COLLADA file as an external scene. Each external scene contains cameras, lights, instanced geometries, skeletal animations, curves, animations and so on. Vanda Engine imports static and dynamic cameras, static and dynamic lights, static and dynamic instanced geometries, and skeletal animations including their animation clips.

Does Vanda Engine let me add multiple external scenes to the current

VScene?

Yes, Each VScene (Vanda Engine Scene) may contain multiple external scenes in COLLADA format. For example, you can export 2 different scenes from 3ds Max, 3 different scenes from Cinema 4D, and add them to the current VScene via the import menu. Each external scene is rendered in its own layer.

Does Vanda Engine let me import multiple PhysX scenes in XML format?

No. Each VScene contains one active PhysX scene. Vanda Engine lets you import one external PhysX Scene in XML format.

I haved imported a PhysX scene in XML format. Why Vanda Engine does not generate PhysX actors for selected instanced geometries and skins?

Vanda Engine supports 2 types of physics scenes:

- PhysX scenes imported from 3D tools capable of exporting to PhysX XML format.
 - PhysX scenes generated inside Vanda Engine.

Each VScene contains one active PhysX scene. Built in PhysX system of Vanda Engine works with PhysX scenes generated inside Vanda Engine.

Made with help of Dr.Explain