Candera Engine 3D

Uniformbuffers for Light and Material

Candera Light and Material parameters are now passed as uniform buffers to shaders whenever possible. The Candera reference shader include files have been updated so that any existing shader using Candera's Light and/or Material automatically benefits from this new feature without any change necessary. The Renderer::Statistics has been enhanced by the number of instantiated uniform buffer objects and the number of uniform buffer objects that were updated.

Renderer Statistics Overlay

This feature enables the user to draw renderer statistics data into a render target for diagnostic purposes. A common case would be to draw it every frame after rendering all objects so that it constitutes an overlay presenting relevant rendering data to the user in realtime. The color of the overlay's text and values can be configured separately, as well es the properties of the appearance used to draw the overlay. Added the ability to draw only parts of a VertexBuffer to RenderDevice.

See also:

RendererStatisticsOverlay.

New 3D Animation Property Setters

New animation property setters have been implemented for the Viewport (Camera::GetViewport) and ScissorRectangle (Camera::GetScissorRectangle) properties of a Camera object from a 3D Scene. The corresponding classes for these are Animation::CameraViewportPropertySetter and Animation::CameraScissorRectanglePropertySetter.

A new animation property setter has been implemented also for the vertex rectangle property of renderable canvas objects. The corresponding class for this is Animation::CanvasVertexRectanglePropertySetter.

Transform Feedback

Candera 3.4.0 supports transform feedback on OpenGL ES3.0 platforms. This feature enables the user to utilize the vertex processing unit of the GPU to perform calculations and reuse the results. This can be used for debugging vertex transformations, speed up particle systems, or reuse skinned mesh data for physics calculations.

Pixel Buffer Objects

Candera 3.4.0 supports pixel buffer objects on OpenGL ES3.0 platforms. This feature can be used transfer data to and from video memory asynchronously. The Renderer::Statistics has been enhanced by the number of instantiated pixel buffer objects.

See also:

Renderer::UploadPixelBuffer.

Submesh

This version introduces the ability to draw a subset of a vertex buffer.

See also:

VertexBuffer::SetElementStart and VertexBuffer::SetElementCount.

RenderMode Rasterizer Discard

The RenderMode now offers an option to discard rasterization. This means that vertex shaders are processed while fragment shaders are not. This is useful in conjunction with transform feedback. This feature is OpenGL ES3.0 only.

See also:

RenderMode::SetRasterizerDiscardEnabled.

Hashes for RenderOrderBin Comparison

During runtime, the RenderOrderBins have been compared via String compare, which is very time consuming, especially if long names are used that differ only in the last letters. To reduce CPU time, hashes are used now (instead of IDs in order to not break the existing interface).

See also:

Node::GetRenderOrderBinAssignmentHash.

Revision #2
Created 2023-03-02 01:37:27 UTC by Kanai Tomoaki
Updated 2023-06-13 12:20:25 UTC by Kanai Tomoaki