Render Loop

Description

After loading an asset and creating scenes, this chapter will now explain how an application can trigger and control rendering of 3D and 2D scenes.

Basically it is up to the application to define an event loop triggering render calls. Candera provides interfaces to render specific or all cameras, but scheduling and triggering these render calls is what the application is responsible for.

3D Render Loop

3D Render Calls

Static class Candera::Renderer is part of Candera 3D Engine and provides two render call interfaces:

1. Render all 3D cameras: Candera::Renderer::RenderAllCameras
2. Render a specific camera only: Candera::Renderer::RenderCamera

Usually, rendering of all cameras is the safer and most convenient way, hence chosen here for the tutorial:

```
    FEATSTD_LINT_NEXT_EXPRESSION(534, "return value not needed");
    Renderer::RenderAllCameras();
```

Render Buffer Swapping

If the draw target operates on multiple buffers after each rendering cycle the presentation buffer is exchanged with the current background buffer, on which was written during the cycle.

For this, three interfaces are available:

- Candera::RenderTarget::SetAutoSwapEnabled(bool enabled): if *true*, render target buffers are swapped automatically, when a camera within the render target has been rendered. Default: *false*

- Candera::Camera::SetSwapEnabled(bool enable): if *true*, buffers are swapped automatically after rendering the according camera. Default: *false*
- Candera::RenderTarget::SwapBuffers(): Trigger buffer swapping 'by hand'

It is very important to control buffer swapping in the application correctly:

- If swapping is never done, the display will stay black
- If buffers are swapped too often, the display will flicker.

The application doesn't need to swap its render targets in case:

- the render target is set to swap automatically. For this, set the render target property "SetAutoSwapEnabled" to *true*.
- the proper camera is set to swap automatically. For this, set the camera property "SwapEnabled" to *true*.

> If you use more than one camera on a render target, take care to enable swapping only for the last camera (highest sequence number).

2D Render Loop

2D Render Calls

2D Render Call interfaces are provided by Candera::Renderer2D.

This is the 2D specific render call equivalent to 3D.

```
    Renderer2D::RenderAllCameras();
```

Render Buffer Swapping

For 2D, render buffer swapping is analogous to 3D: The default value of 2D RenderTarget property "AutoSwapEnabled" and 2D Camera property "SwapEnabled" is *false*.

So take care to

- either swap render target
- or swap one of the cameras used
- or trigger buffer swapping on the render target 'by hand'.

Revision #7
Created 2023-05-01 01:27:11 UTC by Tsuyoshi.Kato
Updated 2025-01-17 08:54:40 UTC by Elisabeth Zauner