Finalization

Description

This tutorial explains how to shut down a 2D or 3D application safely.

Releasing Scenes

First all 3D or 2D scenes must be released.

- 3D Scenes:

```
    contentLoader->ReleaseAllSceneContexts();
    m_sceneContexts.Clear();
```

- 2D Scenes: The same procedure as for 3D Scenes, whereas the Candera::ContentLoader interface for releasing 2D scene contexts is different:

```
    contentLoader->ReleaseAllScene2DContexts();
    m_sceneContexts.Clear();
```

Don't forget to shutdown the TextEngine, if there is one in use:
`Candera::TextRendering::FontEngine::Instance().Shutdown();`

Releasing AnimationPlayer

Next, all used Candera::AnimationPlayer and Candera::AnimationPlayerGroup instances have to be removed from the Candera::AnimationTimeDispatcher and freed.

Unloading RenderTargets and Displays

All involved render targets / layers need to be unloaded from video memory:

```
    if (m_isRenderTargetsUploaded) {
        for (Int index = FeatStd::Internal::NumericConversion<Int>(m_gdus.Size()) - 1; index >=
            GraphicDeviceUnit* gdu = m_gdus[index];
            if (gdu != 0) {
                if (0 != gdu->ToRenderTarget3D()) {
                    static_cast<void>(gdu->ToRenderTarget3D()->Activate());
```

```
                }
                gdu->Unload();
            }
            m_gdus[index] = 0;
        }
    }
    m_gdus.Clear();
```

Destory Displays in use:

```
    FEATSTD_LOG_INFO("- Destroy displays ...");
    Vector<Display*>::Iterator it;
    for (it = m_displays.Begin(); it != m_displays.End(); ++it) {
        Display* display = *it;
        if (display != 0) {
            display->Unload();
            DevicePackageInterface::DestroyDisplay(display);
        }
    }
    FEATSTD_LOG_INFO("- done");
    m_displays.Clear();
```

Finalizing AssetProvider

Finally, Candera::AssetProvider and Candera::ContentLoader shall be finalized:

```
    m_assetProvider.Finalize();
    m_contentLoader = 0;
```

Destructing Other Resources

Don't forget to destruct any other resources in the application destructor.

Revision #5
Created 2023-05-01 01:14:15 UTC by Tsuyoshi.Kato
Updated 2024-07-31 23:18:32 UTC by Tsuyoshi.Kato