Candera V3.3.1

Release Date: February 2017

To keep backwards compatibility for any interface change in the public API the old interface has been kept and marked as deprecated. The deprecated interfaces will be removed in future versions. Compiler warnings will be generated when using a deprecated interface through the usage of Candera Macro CANDERA_DEPRECATED_3_3_1.

- Candera/FeatStd CMake Build System Changes
- Candera Engine 3D
- Candera Engine 2D
- Candera Base
- Candera Monitor
- Candera Device
- Candera AssetLoader
- TextEngine
- Candera Behavior
- Migration Guide

Candera/FeatStd CMake Build System Changes

New CMake variables were added:

Name	Description	Possible values
CANDERA_TEXTENGINE_WORKER_THREAD_ENABLED	Uses a worker thread for asynchronous text rendering when enabled. Otherwise it uses a single threaded solution.	ON/OFF (Default: ON)

Candera Engine 3D

Glyph Atlas TextNode Renderer

This version introduces a new text node renderer for TextNode2D which uses a glyph atlas for rendering.

Candera Engine 2D

Glyph Atlas TextBrush Cache

This version introduces a new text brush cache which uses a glyph atlas.

Reworked TextNode2D

Parts of TextNode2D have been rewritten. Main changes happened to the internal render processes. The rework was necessary for asynchronous text rendering. The performance of TextNode2D increased with this rework.

The behavior of TextNode2D slightly changed:

- It were possible to attach a TextNode2DLayouter to TextNode2D and not using the layouter. This led to undesired visual output but text were visible. Now in this case the layouter has to be used otherwise no (or old) text will be shown.
- In some corner cases (e.g. mixed bidirectional text) the truncation with ellipsis can truncate the text in a wrong manner. This will be part of the next TextEngine update.
- Fixed some incorrect layout issues.
- Some of these layout issues affect all text types which are using TextEngine (not only TextNode2D).
- A multiline text with a truncated line did not show all the lines after the truncated line anymore. This has been fixed.

Asynchronous text rendering

TextNode2D can be rendered asynchronously now. A SceneComposer flag for each TextNode2D were added. With this update text can be rendered in a worker thread or in a single threaded version. This improves the handling of the resources used to render text. Performance of text rendering might be improved on some systems (e.g. multicore system or using the vsync time to render).

Further information of how to use asynchronous text rendering can be found in the Text-Tutorial (No. 5) which received a new chapter for asynchronous text rendering.

Candera Base

Bitmap Atlas

The BitmapAtlas class packs several smaller images into the bitmap of the atlas. It supports padding between packed images so that the user can mipmap the atlas if necessary.

Glyph Atlas

The GlyphAtlas stores glyphs in bitmap atlases together with lookup information for re-use.

Candera Monitor

The monitor and analyzer supports recording performance logs in multiple threads now. It is advised to use the synchronization barrier (macros in PerfMonPublicIF.h) when dumping data. However, it is not necessary in most cases.

It is also advised to use different recorder for each thread. Recorder which do not have an over time information (e.g. ValueRecorder or EventRecorder) are excluded from this restriction. All the other recorder types may work but depends on the use case.

Candera Device

Support for Device Specific Properties in Generic OpenGL ES Device Package

Device packages based on Generic OpenGL ES Device Package can now extend the common Window properties with device specific ones, like special color formats or swap behaviors for example.

Candera AssetLoader

Asynchronous asset loading

Assets can be loaded asynchronously now. A new CMake flag CANDERA_ASSETLOADER_WORKER_THREAD_ENABLED specifies if the asynchronous asset load requests are handled on a dedicated thread or on the caller one. To load the assets asynchronously, a new interface was added - AsyncAssetProviderProxy - available from any AssetProvider implementation via the AssetProvider::AsyncProxy() method.

TextEngine

The TextEngine had two layout issues: Justified alignment were not always correctly left or right aligned. Another effect of this issue was that glyphs at the border moved outside of the defined text area. This has been fixed.

The text were not correctly aligned when word wrap is disabled and the available text area is smaller than a line of text. E.g. A centered text kept being centered, so only the middle of the text were visible. The text is now left or right aligned if the line is longer than the available size.

Candera Behavior

Layout Invalidation

A convenience function to trigger Layout Invalidation was added.

Migration Guide

Any Candera::TextRendering::FontStore derived classes need to add a call to Candera::GlyphAtlas::GetInstance().RemoveFont(Candera::TextRendering::FaceIdentifier) in their Unload function.

Following an overview about interface changes between Candera V3.3.0 and Candera V3.3.1.

Description	Candera V3.3.0	Candera V3.3.1