Offscreen Rendering

Overview

Offscreen rendering lets you obtain the content of a `BrowserWindow` in a bitmap, so it can be rendered anywhere, for example, on texture in a 3D scene. The offscreen rendering in Electron uses a similar approach to that of the Chromium Embedded Framework project.

Notes:

- There are two rendering modes that can be used (see the section below) and only the dirty area is passed to the `paint` event to be more efficient.
- You can stop/continue the rendering as well as set the frame rate.
- The maximum frame rate is 240 because greater values bring only performance losses with no benefits.
- When nothing is happening on a webpage, no frames are generated.
- An offscreen window is always created as a Frameless Window..

Rendering Modes

GPU accelerated GPU accelerated rendering means that the GPU is used for composition. Because of that, the frame has to be copied from the GPU which requires more resources, thus this mode is slower than the Software output device. The benefit of this mode is that WebGL and 3D CSS animations are supported.

Software output device This mode uses a software output device for rendering in the CPU, so the frame generation is much faster. As a result, this mode is preferred over the GPU accelerated one.

To enable this mode, GPU acceleration has to be disabled by calling the [`app.disableHardwareAcceleration()`][disablehardwareacceleration] API.

Example

“‘javascript fiddle=‘docs/fiddles/features/offscreen-rendering’ const { app, BrowserWindow } = require(‘electron’) const fs = require(‘fs’)

app.disableHardwareAcceleration()

let win

app.whenReady().then(() => { win = new BrowserWindow({ webPreferences: { offscreen: true } })

win.loadURL(‘https://github.com’) win.webContents.on(‘paint’, (event, dirty, image) => { fs.writeFileSync(‘ex.png’, image.toPNG()) }) win.webContents.setFrameRate(60) }) “‘

After launching the Electron application, navigate to your application's working folder, where you'll find the rendered image. [disablehardwareacceleration]: ../api/app.md#appdisablehardwareacceleration