

ELECTRON PRO-TIPS™

@PAULCBETTS (GITHUB, TWITTER)

HI.

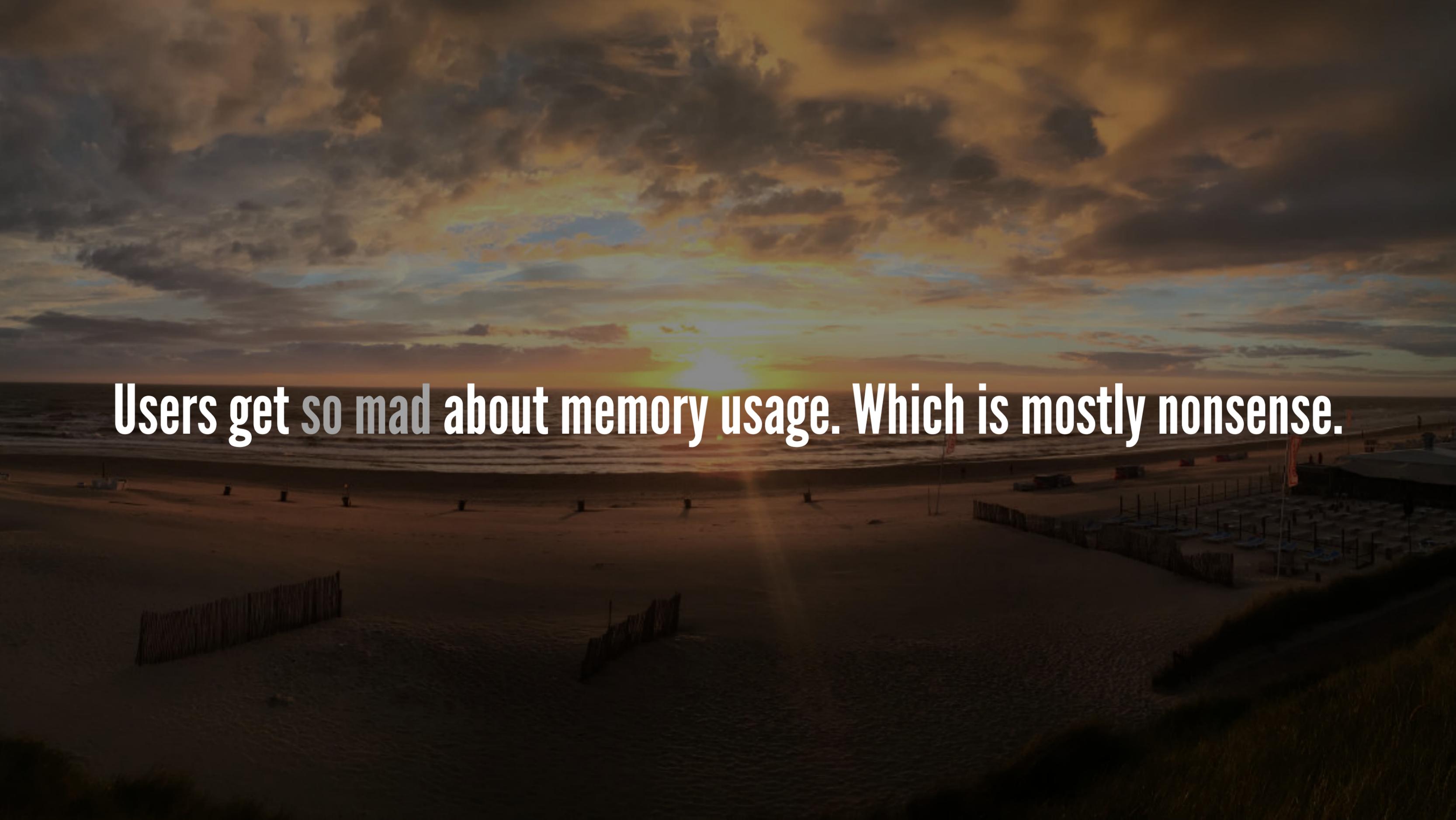
HERE'S A FEW THINGS I NOTICE PEOPLE DOING IN ELECTRON APPS

(THAT MAKE USERS MAD)

A wide-angle photograph of a beach at sunset. The sun is low on the horizon, casting a warm, golden glow across the sky and the water. The sky is filled with soft, dark clouds. The beach is visible in the foreground, with some structures and a fence line. The overall mood is serene and contemplative.

MEMORY USAGE

MATTERS

A wide-angle photograph of a beach at sunset. The sky is filled with dramatic, dark clouds illuminated from below by the setting sun, creating a gradient of orange, yellow, and blue. The sun is positioned just above the horizon line, casting a long, shimmering reflection across the water. In the foreground, a large, flat expanse of sand is marked with a grid of vertical wooden stakes, likely for a construction or agricultural project. The overall mood is serene yet somber due to the low light.

Users get so mad about memory usage. Which is mostly nonsense.

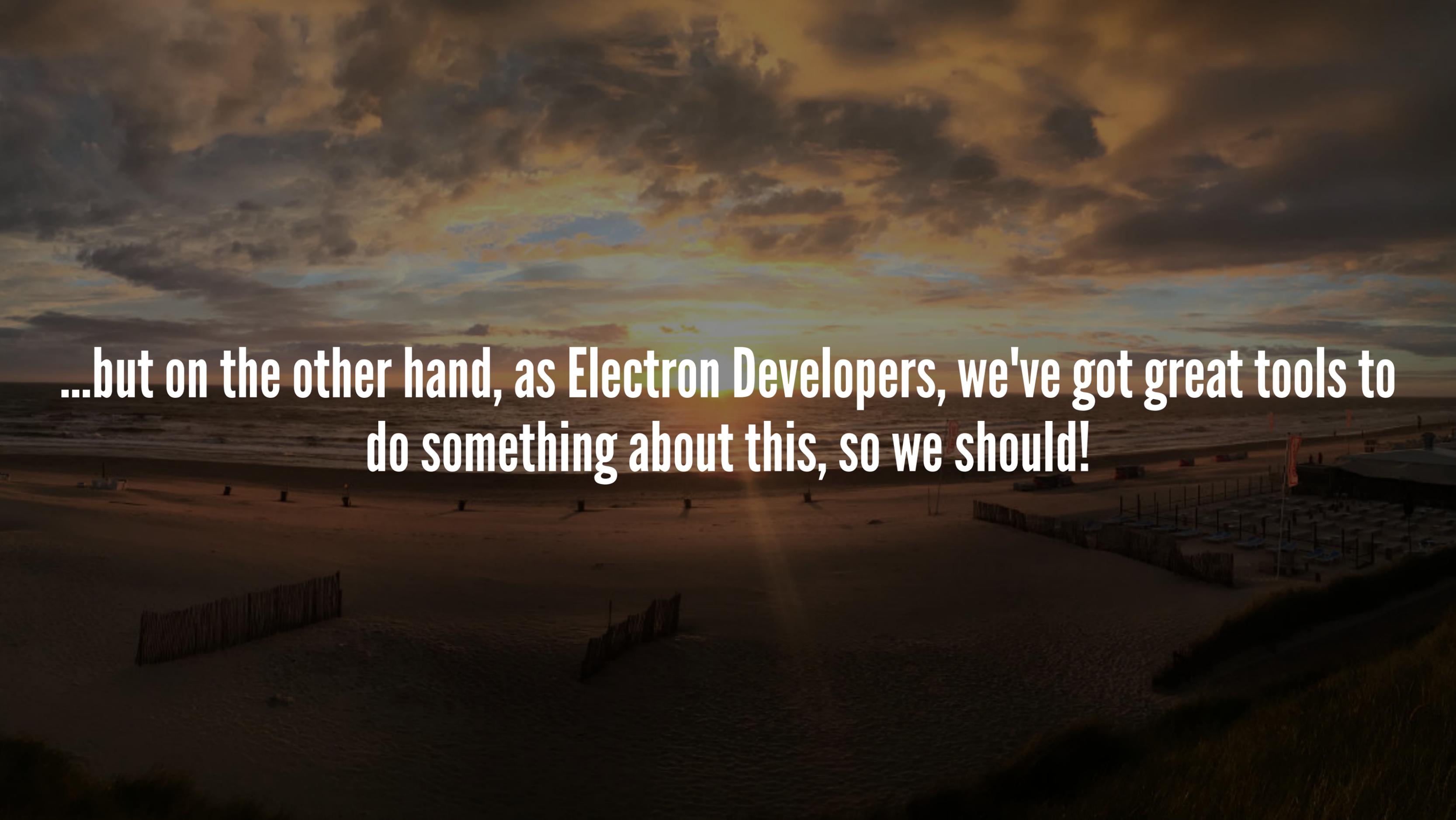
Every conversation I've ever had about Electron memory usage:

Them: IM SO MAD ABOUT MEMORY USAGE

Me: I understand! So, what's the "Commit Charge" say in Task Manager? That's the percentage of RAM that is actually in-use.

Them: Oh, it's 40%.



A wide-angle photograph of a beach at sunset. The sky is filled with dramatic, dark clouds illuminated from below by the setting sun, creating a warm orange and yellow glow. The sun is visible as a bright point of light on the horizon. In the foreground, the sandy beach is dark, with several wooden posts or markers scattered across it. To the right, there is a construction site with a grid of rebar and concrete forms, suggesting a building or structure is being built. The overall mood is contemplative and hopeful.

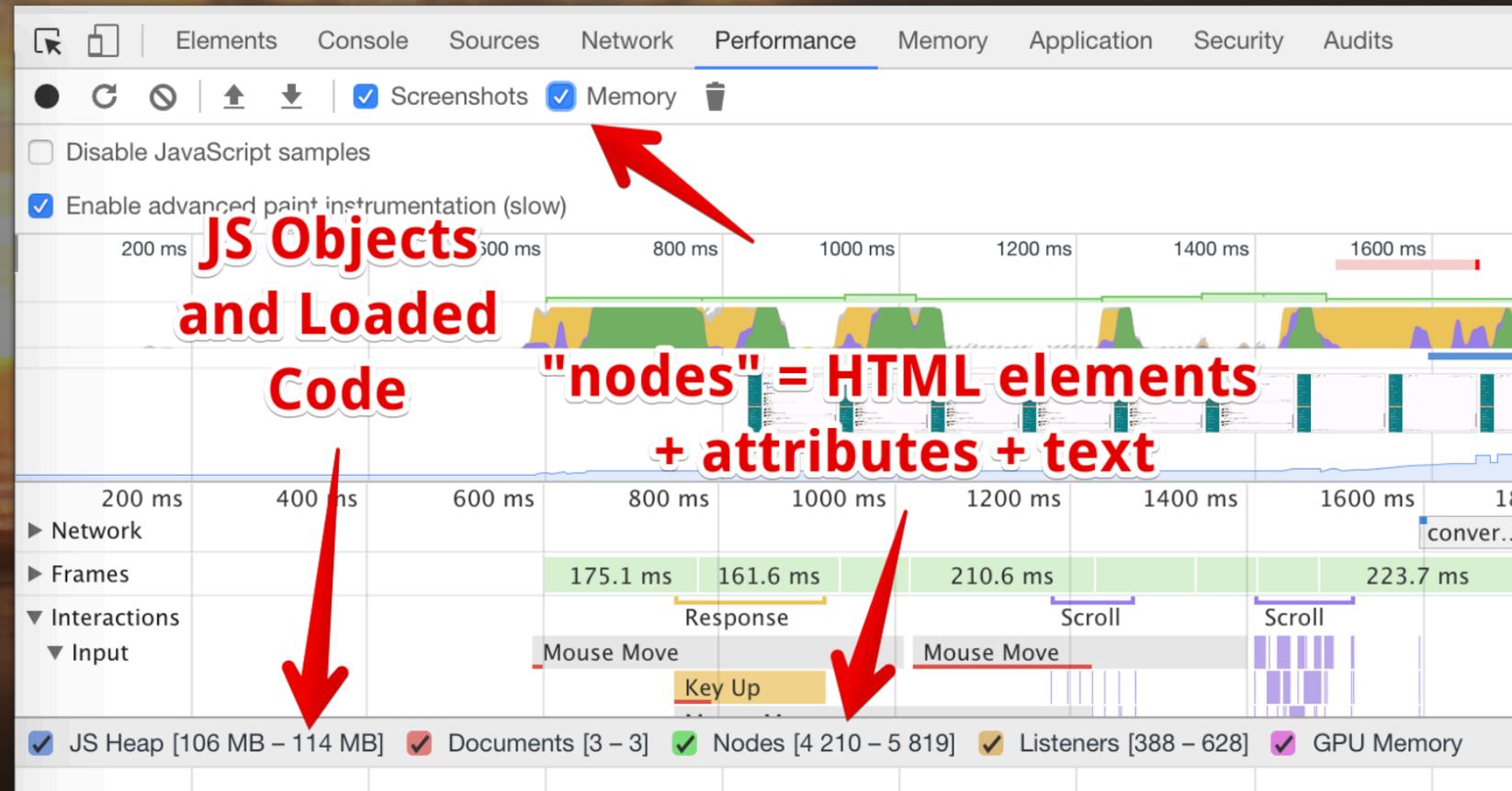
...but on the other hand, as Electron Developers, we've got great tools to do something about this, so we should!

A wide-angle photograph of a beach at sunset. The sky is filled with dramatic, dark clouds illuminated from below by the setting sun, creating a warm orange and yellow glow. The sun is visible as a bright point on the horizon over the ocean. The beach in the foreground is dark, with several wooden fences or barriers visible. The text 'LOAD LESS STUFF' is overlaid in large, white, bold, sans-serif capital letters across the center of the image.

LOAD LESS STUFF

LOAD LESS STUFF

- ▶ Lots and Lots of DOM Elements
 - ▶ Especially Images
 - ▶ JS Heap



A sunset over a beach with a large grid of lounge chairs in the foreground. The sky is filled with soft, golden light and scattered clouds. The ocean is visible in the distance, and the foreground shows a sandy beach with a grid of lounge chairs and some orange flags.

USE REACT OR

VUE

(AND VIRTUALIZING LISTS)

LOAD LESS STUFF

- ▶ Libraries that you load in your app never get unloaded
- ▶ Bad for startup performance and for memory usage!

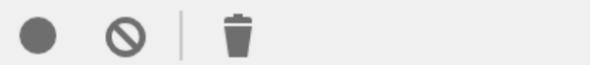
**require
takes time!**



(anonymous)	(anonymous)	(anonymous)	(anonymous)
startup	startup	startup	startup
Module.runMain	Module.runMain	Module.runMain	Module.runMain
_tickCallback	_tickCallback	_tickCallback	_tickCallback
(anonymous)	(anonymous)	(anonymous)	(anonymous)
Promise.resolve.then	Promise.resolve.then	Promise.resolve.then	Promise.resolve.then
require	require	require	require
Module.require	Module.require	Module.require	Module.require
Module._load	Module._load	Module._load	Module._load
tryModuleLoad	tryModuleLoad	tryModuleLoad	tryModuleLoad
Module.load	Module.load	Module.load	Module.load
require.extensions.(anonymous function)	require.extensions.(anonymous function)	require.extensions.(anonymous function)	require.extensions.(anonymous function)
Module._compile	Module._compile	Module._compile	Module._compile
(anonymous)	(anonymous)	(anonymous)	(anonymous)
(anonymous)	(anonymous)	(anonymous)	(anonymous)
require	require	require	require
Module.require	Module.require	Module.require	Module.require
Module._load	Module._load	Module._load	Module._load
tryModuleLoad	tryModuleLoad	tryModuleLoad	tryModuleLoad
Module.load	Module.load	Module.load	Module.load
require.extensions.(anonymous function)	require.extensions.(anonymous function)	require.extensions.(anonymous function)	require.extensions.(anonymous function)
Module._compile	Module._compile	Module._compile	Module._compile
(anonymous)	(anonymous)	(anonymous)	(anonymous)
(anonymous)	(anonymous)	(anonymous)	(anonymous)
require	require	require	require

A wide-angle photograph of a beach at sunset. The sky is filled with dark, dramatic clouds, with a bright orange and yellow glow from the setting sun just above the horizon. The ocean is visible in the middle ground, and the sandy beach is in the foreground. In the distance, there are some structures and what appears to be a parking lot or a fenced-in area. The overall mood is serene and atmospheric.

USE THE HEAP PROFILER



Summary All objects

Profiles

HEAP SNAPSHOTS

Snapshot 1
15.4 MB [Save](#)

Constructor	Distance	Objects Count	Shallow Size	Retained Size
▶ global / file://	1	1 0 %	56 0 %	12 887 019 80
▶ (compiled code)	3	10 321 9 %	2 040 872 13 %	10 506 120 65
▶ Object	2	1 376 1 %	81 352 1 %	7 247 528 45
▶ (string)	2	13 094 11 %	6 911 432 43 %	6 911 432 43
▶ (system)	-	61 251 52 %	2 675 056 17 %	3 961 104 25

**DON'T RUN STUFF
IN THE MAIN
PROCESS**

A scenic landscape photograph taken during the "blue hour" of sunset or sunrise. The sky transitions from a deep blue at the top to a warm orange glow near the horizon. In the foreground, a large, textured tree trunk stands on the right side. The middle ground is dominated by a rocky, light-colored outcrop. The background shows a vast, hazy mountain range under a soft, golden light. The text "but what about...?" is overlaid in the center in a white, sans-serif font.

but what about...?

NO



WHAT THE MAIN PROCESS IS NOT

- ▶ "The Backend"
- ▶ "A Background Thread"
- ▶ "The Server"



**THE MAIN
PROCESS IS FOR
ORCHESTRATION**

THE MAIN PROCESS IS FOR ORCHESTRATION

Running code in the main process slows everything else down

Chromium uses IPC internally to do things, such as signaling window size changes

So when the main thread is busy, your app glitches!

THE MAIN PROCESS IS FOR ORCHESTRATION

`ipc.send` is asynchronous which is Better, but not enough!

The main process can still do a lot of work as a result of `ipc.send`, and
block stuff

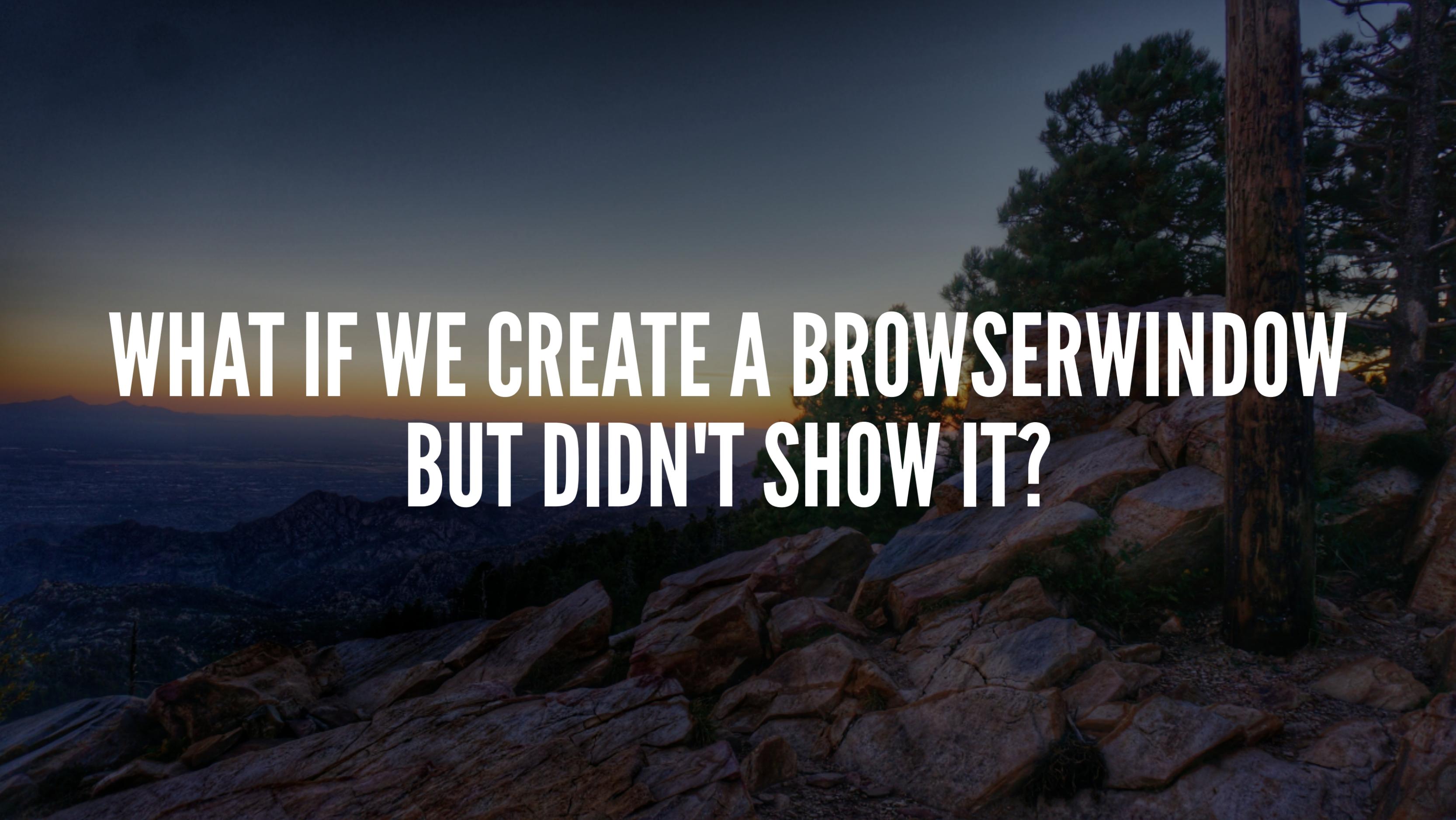
THE MAIN PROCESS IS FOR ORCHESTRATION

The main process should really only be used to tell other processes what to do

- ▶ Sending information between windows
- ▶ Signalling menu items and dock events
- ▶ Crash reporting and other APIs that only work in the main process



**SO HOW CAN I DO
STUFF THEN??**

A scenic view of a mountain peak at dusk. The foreground is dominated by large, layered rock formations. A large, dark tree trunk stands on the right side. In the background, a valley is visible under a twilight sky with a soft orange glow on the horizon.

**WHAT IF WE CREATE A BROWSERWINDOW
BUT DIDN'T SHOW IT?**



Search or jump to...

Pull requests Issues Marketplace Explore



electron-userland / electron-remote

Unwatch 12

Star 366

Fork 30

Code Issues 15 Pull requests 4 Projects 0 Wiki Insights Settings

Execute JavaScript in remote electron browser with remote server

Edit

ELECTRON-REMOTE

Manage topics

135 commits

2 branches

2 releases

5 contributors

MIT

Branch: master

New pull request

Create new file

Upload files

Find file

Clone or download

paulcbetts Merge pull request #41 from electron-userland/security-upgrades

Latest commit b314f9c on Jun 12

src Add configurable timeouts to several things 5 months ago

test Update the tests a bit 5 months ago

ELECTRON-REMOTE, DOING WORK FROM THE MAIN PROCESS

```
import { createProxyForRemote } from 'electron-remote';

// myWindowJs is now a proxy object
// for myWindow's `window` global object
const myWindowJs = createProxyForRemote(myWindow);

// Functions suffixed with _get
// will read a value
userAgent = await myWindowJs.navigator.userAgent_get()
```

window.requestIdleCallback()

Jump to: [Syntax](#) [Example](#) [Specifications](#) [Browser compatibility](#) [See also](#)

Web technology for developers >

Web APIs (HTML, CSS, JS)

window.requestIdleCallback()

Related Topics

Window

▼ Properties

`applicationCache`

🔍 `caches`

`closed`

`console`

`controllers`

`crypto`

`customElements`

⚠️ This is an experimental technology

Check the Browser compatibility table carefully before using this in production.

The `window.requestIdleCallback()` method queues a function to be called during a browser's idle periods. This enables developers to perform background and low priority work on the main event loop, without impacting latency-critical events such as animation and input response. Functions are generally called in first-in, first-out order; however, callbacks which have a higher priority may be called out-of-order if necessary in order to run them before the next release.

You can call `requestIdleCallback()` within an idle callback function to schedule another callback to take place no sooner than the next pass through the event loop.

📄 A `timeout` option is strongly recommended for required work, as otherwise it's possible multiple seconds will elapse before the callback is fired.

REQUESTIDLECALLBACK IS SUPER COOL

Like `setTimeout` but only runs once the UI is no longer busy

The callback allows you to repeatedly schedule `requestIdleCallback` to do work in a loop

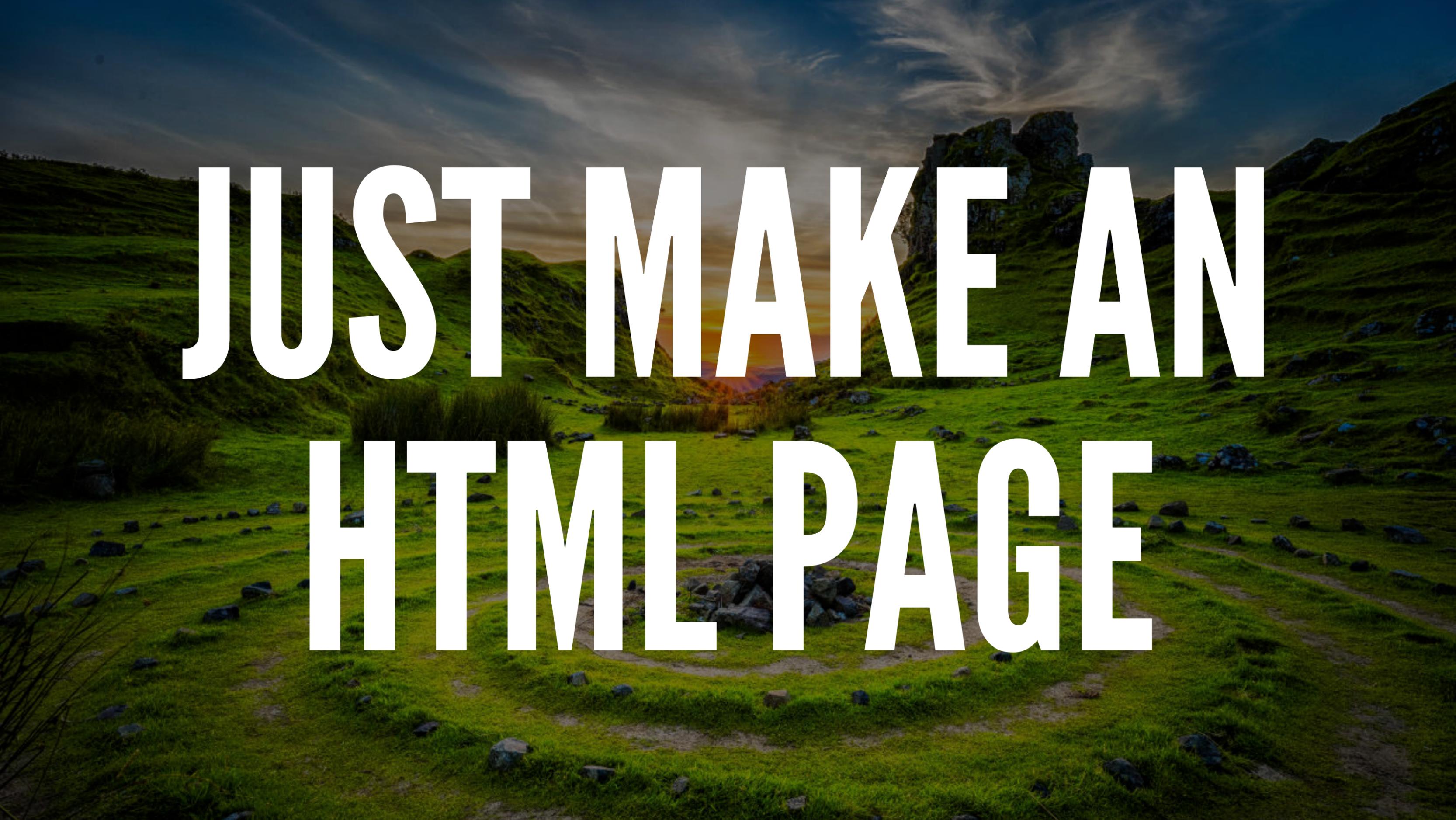
Writing App Data in the background is a great place to use `requestIdleCallback`

ELECTRON-REMOTE, TASKPOOL

```
import { requireTaskPool } from 'electron-remote';

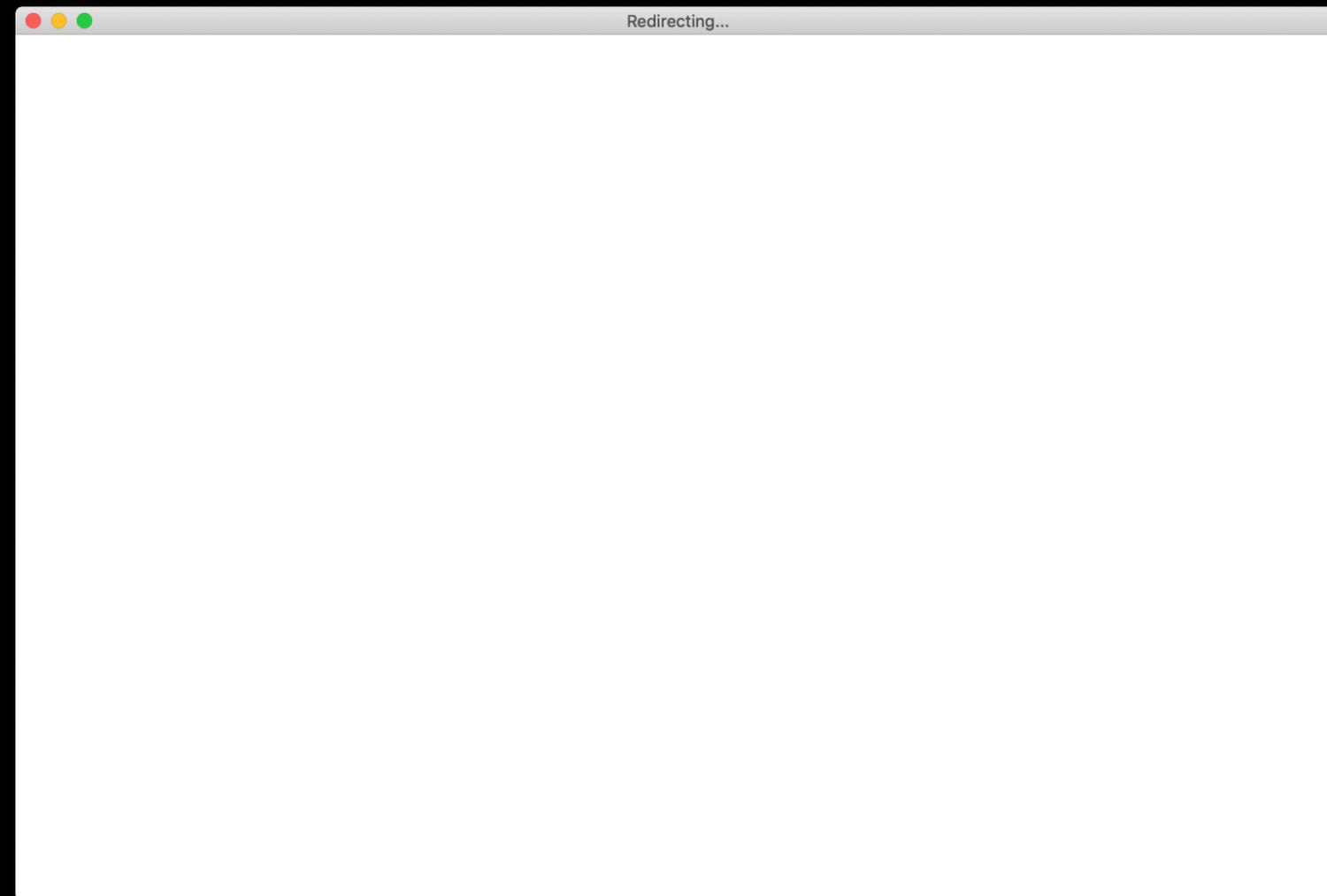
const myCoolModule = requireTaskPool(
  require.resolve('./my-cool-module'));

// This method will run synchronously,
// but in a background BrowserWindow process
// so that your app will not block
let result = await myCoolModule.calculateDigitsOfPi(100000);
```

A scenic landscape featuring rolling green hills under a dramatic, cloudy sky at sunset. A prominent, jagged rock formation stands in the middle ground. The foreground is a grassy field with scattered dark rocks and a dirt path. The overall mood is serene and natural.

**JUST MAKE AN
HTML PAGE**

THIS EXPERIENCE IS A DRAG.



JUST MAKE AN HTML PAGE

Putting a website into an Electron frame is easy, but not great for Users

Offline Mode is way easier

Your app will start Really Fast

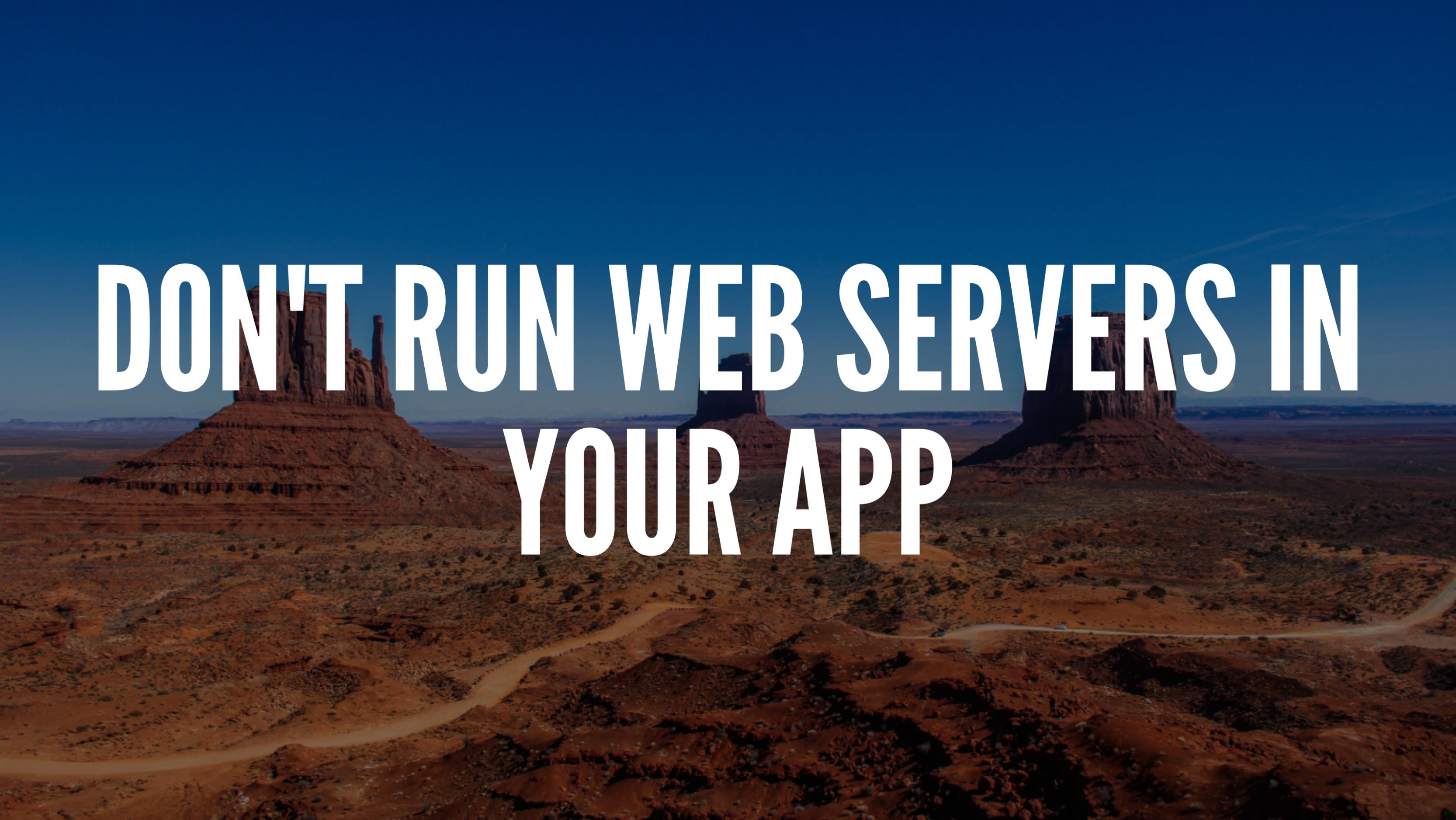
Starting with a Desktop Mindset will make your app feel like an app

HTML PAGES ARE MORE SECURE

Designing a hybrid app is Very Security Sensitive, so that you don't accidentally give Desktop Powers™ to remote content

When all of the code for your app is local, you remove this possibility altogether

XSS is still extremely important to watch out for!

A wide-angle photograph of a desert landscape, likely Monument Valley, featuring several large, iconic butte rock formations. The terrain is arid and brown, with a winding dirt road visible in the foreground. The sky is a clear, deep blue. The text is overlaid in the center of the image.

**DON'T RUN WEB SERVERS IN
YOUR APP**

DON'T RUN WEB SERVERS IN YOUR APP

...cause like, what if more than one user uses your app?

Your web service now a great way to move data between different users

If you run as Admin, it's now a great way to local EoP

...or if you're really unlucky, have arbitrary websites run Desktop code

Electron Forge

The command line interface for ambitious Electron applications

USE ELECTRON-FORGE

```
# Launch your app  
$ cd my-new-project  
$ electron-forge start
```

Ready for a closer look? [Dive into the CLI documentation](#)

Opinionated Electron development

Modern language compilation, one-step builds for all platforms,
and core templates for your favorite frameworks

USE ELECTRON-FORGE

electron-forge handles all of the things you might want to use Express or Webpack for, like Hot Module Reload

It handles Babel/TypeScript/LESS/Sass via hooking Electron and compiling on-the-fly during development

electron-forge does all of the packaging and compilation work too

An aerial photograph of Monument Valley, showing the iconic butte formations of the Navajo Nation. The landscape is a vast, arid desert with reddish-brown soil and sparse vegetation. The sky is a clear, deep blue. The text "BUT I LIKE WEBPACK!" is overlaid in large, white, bold, sans-serif capital letters across the center of the image.

BUT I LIKE WEBPACK!

BUT I LIKE WEBPACK!

Trying to interact with Electron itself gets Weird because now there are two separate module systems

Native node modules are a pain with Webpack, both at runtime and on the build side

Packaging becomes way more complex

PERFORMANCE,
BLAHHHH





SECURITY, BORINGGGGG

MEMORY, UGHHHHHHHH





Microsoft

T6976

NODE-RT IS COOL

Node-RT: Bringing Native Windows Features from Node.js and Electron
Jindav Bala, Senior Software Engineer
Felix Rindberg, Senior Desktop Engineer @ Slack

#MSBuild

Microsoft Build 2017

A photograph of a busy Japanese street at dusk. The street is lined with multi-story buildings, many of which have illuminated signs and advertisements. People are walking on the sidewalks, and some are looking at their phones. The sky is a deep blue, and the streetlights are on. The overall atmosphere is that of a vibrant, modern urban environment.

**CALL WIN10 APIS FROM
ELECTRON SUPER EASILY**

SOME COMPELLING EXAMPLES:

- Windows.Devices.Display
- Windows.Devices.Geolocation
- Windows.Media.Capture
- Windows.Media.OCR
- Windows.System.Power



WHAT ABOUT MACOS?





「ツ」

— _ (ツ) _ / —

There's no easy way to call macOS APIs from Electron, you have to write a Native Node Module.

You can do very simple things with node-ffi, but more complicated things will lead to Segfault City



HOW CAN I FIGURE OUT WHAT
I CAN DO?

Recent

Sort by: Default



Search (Ctrl+E)



Installed

Visual C#

Get Started

Windows Universal

Windows Desktop

.NET Core

.NET Standard

Test

Visual Basic

Visual C++

JavaScript

Other Project Types

Dependency Validation

Online

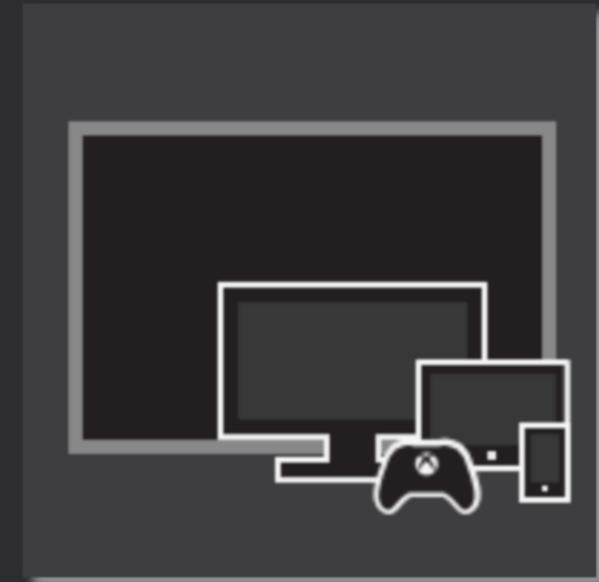
Not finding what you are looking for?

[Open Visual Studio Installer](#)

Icon	Project Name	Type
	Blank App (Universal Windows)	Visual C#
	Class Library (Universal Windows)	Visual C#
	Windows Runtime Component (Universal Windows)	Visual C#
	Optional Code Package (Universal Windows)	Visual C#
	Unit Test App (Universal Windows)	Visual C#
	Coded UI Test Project (Universal Windows)	Visual C#
	Windows Application Packaging Project	Visual C#

Type: Visual C#

A project for a single-page Universal Windows Platform (UWP) app that has no predefined controls or layout.



Name: App1

Location: C:\Users\me\source\repos

Browse...

Solution: Create new solution

Solution name: App1

Create directory for solution

Create new Git repository

OK

Cancel

- Assemblies
- Projects
- Shared Projects
- Universal Windows
 - Core
 - Extensions
 - Recent
- Browse

Filtered to: SDKs applicable to App2

Name	Version
Microsoft General MIDI DLS for Universal Windo...	10.0.171...
Microsoft Universal CRT Debug Runtime	10.0.171...
Microsoft Universal CRT Debug Runtime	10.0.102...
Microsoft Visual C++ 2013 Runtime Package for...	14.0
Microsoft Visual Studio Test Core	15.5
Microsoft Visual Studio Test Core	15.0
MSTest for Managed Projects	15.5
MSTest for Managed Projects	15.0
Visual C++ 2012 UWP Desktop Runtime for nativ...	14.0
Visual C++ 2013 UWP Desktop Runtime for nativ...	14.0
Visual C++ 2015 Runtime for Universal Windows...	14.0
Visual C++ 2015 UWP Desktop Runtime for nativ...	14.0
<input checked="" type="checkbox"/> Windows Desktop Extensions for the UWP	10.0.171...
Windows IoT Extensions for the UWP	10.0.171...
Windows Mobile Extensions for the UWP	10.0.171...
Windows Team Extensions for the UWP	10.0.171...

Search (Ctrl+E)

Name: Windows Desktop Extensions for the UWP

Version: 10.0.17134.0

Targets: UAP 10.0.17134.0

[More Information](#)

Browse... OK Cancel

- Connected Services
- Properties
- References
 - Analyzers
 - Microsoft.NETCore.UniversalWindowsPlatform
 - Universal Windows
- Assets
- App.xaml
- App2_TemporaryKey.pfx
- MainPage.xaml
 - MainPage.xaml.cs
- Package.appxmanifest



App2 - Microsoft Visual Studio

File

Edit

View

Project

Build

Debug

Team

Tools



Solution Explorer



Team Explorer

Ctrl+\, Ctrl+M



Server Explorer



Bookmark Window

Ctrl+K, Ctrl+W



Call Hierarchy



Class View

Ctrl+Shift+C



Code Definition Window

Ctrl+\, D



Object Browser



Error List

Ctrl+\, E



Output



Task List

Ctrl+\, T

- ▷ ■■ Windows.AI.MachineLearning.Preview.MachineLear
- ▷ ■■ Windows.ApplicationModel.Calls.CallsVoipContract
- ▷ ■■ Windows.ApplicationModel.SocialInfo.SocialInfoCo
- ▷ ■■ Windows.ApplicationModel.StartupTaskContract
- ▷ ■■ Windows.Devices.Custom.CustomDeviceContract
- ▷ ■■ Windows.Devices.DevicesLowLevelContract
- ▷ ■■ Windows.Devices.Printers.PrintersContract
- ▷ ■■ Windows.Devices.SmartCards.SmartCardBackground
- ▷ ■■ Windows.Devices.SmartCards.SmartCardEmulatorC
- ▷ ■■ Windows.Foundation.FoundationContract
- ▲ ■■ Windows.Foundation.UniversalApiContract
 - ▷ {} Windows.ApplicationModel
 - ▷ {} Windows.ApplicationModel.Activation
 - ▷ {} Windows.ApplicationModel.AppExtensions
 - ▷ {} Windows.ApplicationModel.Appointments
 - ▷ {} Windows.ApplicationModel.Appointments.Appo
 - ▷ {} Windows.ApplicationModel.Appointments.Data
 - ▷ {} Windows.ApplicationModel.AppService
 - ▷ {} Windows.ApplicationModel.Background
 - ▷ {} Windows.ApplicationModel.Calls
 - ▷ {} Windows.ApplicationModel.Chat
 - ▷ {} Windows.ApplicationModel.Contacts
 - ▷ {} Windows.ApplicationModel.Contacts.DataProvic
 - ▷ {} Windows.ApplicationModel.Contacts.Provider
 - ▷ {} Windows.ApplicationModel.Core
 - ▷ {} Windows.ApplicationModel.DataTransfer
 - ▷ {} Windows.ApplicationModel.DataTransfer.DragD
 - ▷ {} Windows.ApplicationModel.DataTransfer.DragD
 - ▷ {} Windows.ApplicationModel.DataTransfer.ShareT

Assembly **Windows.Foundation.Univers**
C:\Program Files (x86)\Windows Kits\10
\Windows.Foundation.UniversalApiContr

THANKS!

@PAULCBETTS (GITHUB, TWITTER)

ELECTRON PRO TIPS

Red Threads:

- ▶ **Performance And Memory Usage Matters**
- ▶ **Users Care about Memory Usage, so you should too - you have great tools to debug it!**
- ▶ **Just Load Less Stuff - module load time is super easy to debug in Perf tools**