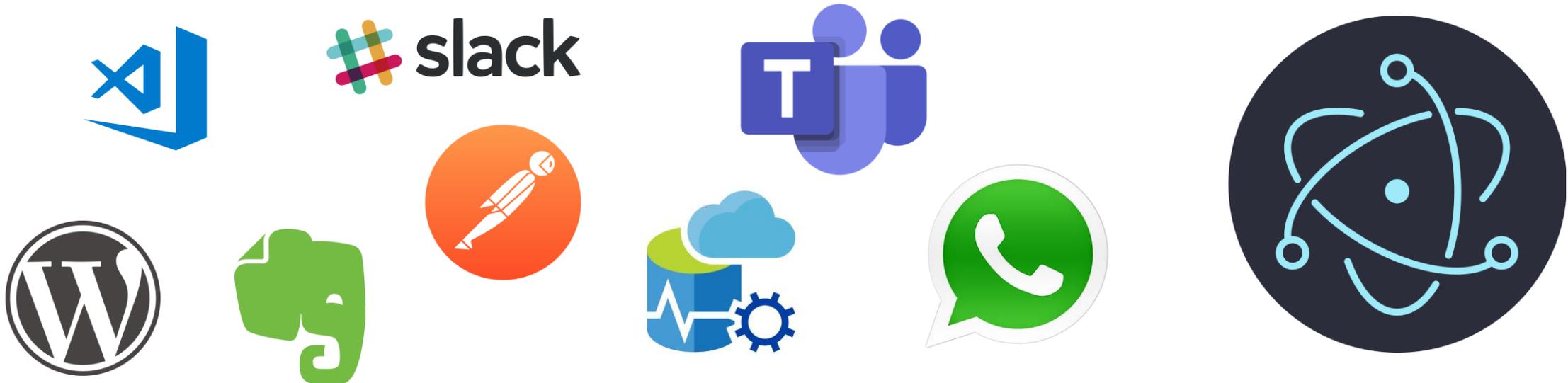# Photino

Create Desktop Apps with Web UIs

Introduction to Photino

Photino is an Open Source project that allows developers to put a WEB UI on a desktop app. Learn how to build apps that run on Windows, Linux and Mac using HTML and your favorite JavaScript framework and/or Blazor for the user interface and .NET 5 for the rest.

- If you're familiar with Electron or Electron applications (VS Code, Slack, Azure Data Studio, Azure Storage Explorer, Evernote…)… Photino is also an Open Source project to allow developers to use Web UIs (HTML, JavaScript, CSS, etc.) in native applications across platforms instead of learning platform specific UI technologies.

The Basics – Photino.Native

- **Photino.Native** is a C++ (and Objective C) wrapper around the OS's built-in Chromium or WebKit-based browser control that opens a native window on the desktop with the browser control in it.
  - On **Windows** this means the **WebView2** control – a browser control for native applications based on Chromium Edge.
  - On **Mac**, this means **WKWebView** – a WebKit browser control for native applications based on Safari.
  - On **Linux**, this means **WebKitGTK+2** – a browser control for native applications based on WebKit directly.

- **Photino.Native** is compiled for each platform and distributed as a NuGet package.

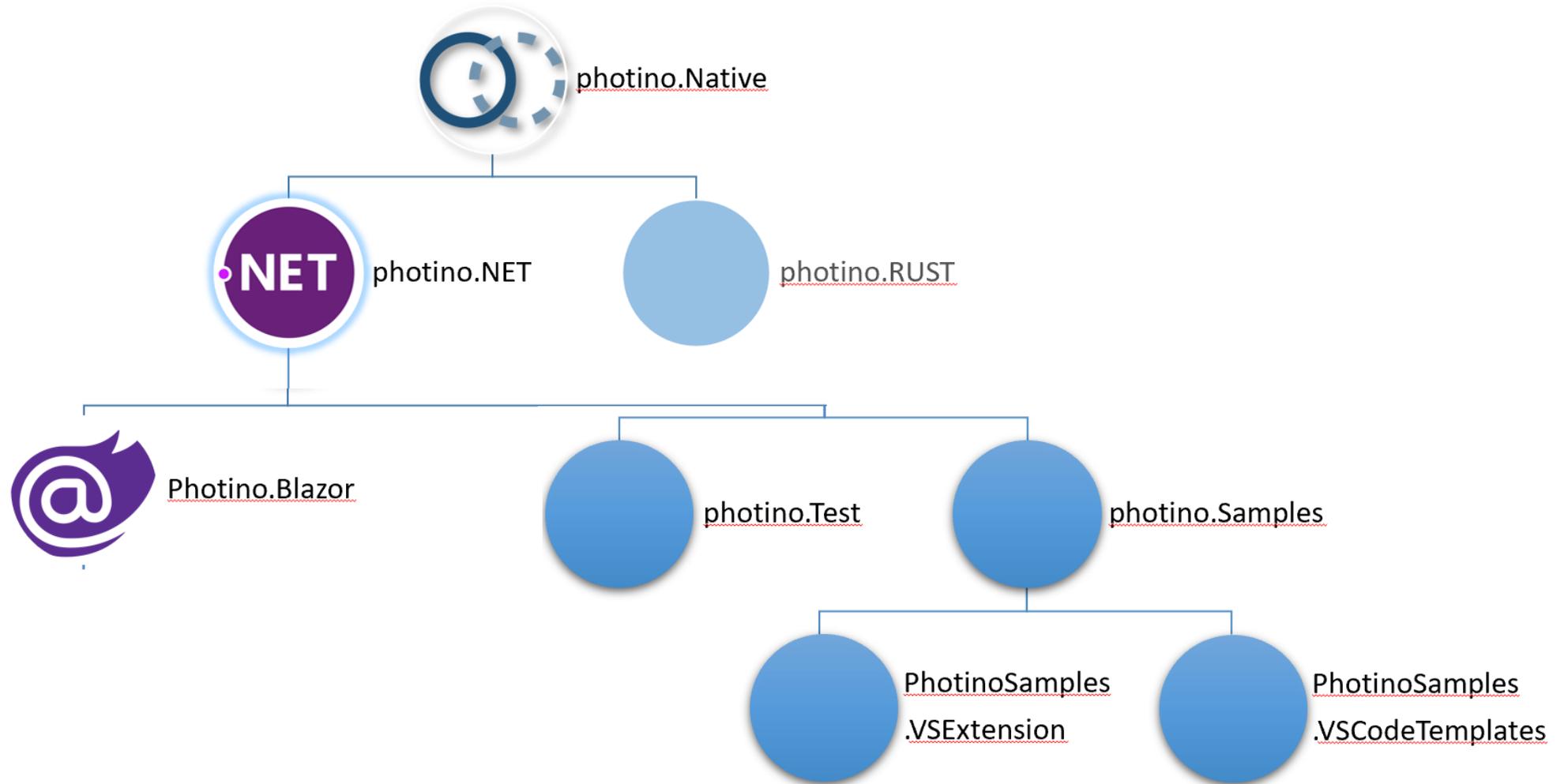# Light Weight & Cross Platform Browser Control

- The Photino.Native wrapper allows us to build desktop applications that are light weight and cross-platform

  - Light weight because the browser control is usually already installed, so we only have to install the wrapper (can be packaged)

  - Cross-platform because applications communicate with the wrapper in exactly the same way on any OS

  - With a cross platform application written in .NET 5, that means 1 code base, 3 Oss... Even for the UI! (.NET 5 can be packaged too!)

Photino.NET

Photino.NET wraps the Photino.Native control and makes it available for .NET developers.

https://www.nuget.org/packages?q=photino

Photino Hierarchy

photino.Native

photino.NET

photino.RUST

Photino.Blazor

photino.Test

photino.Samples

PhotinoSamples
.VSExtension

PhotinoSamples
.VSCodeTemplates
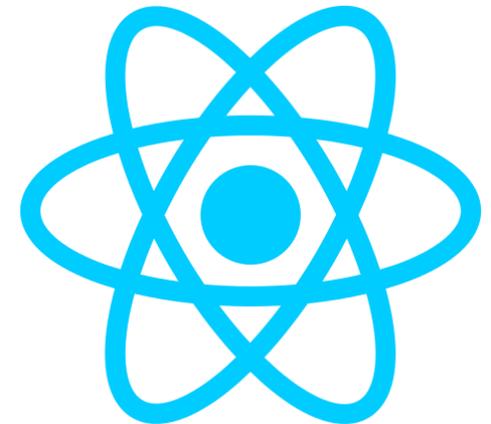
What can I build with Photino.NET?

- HTML, JavaScript, CSS applications

- ReactJS

- AngularJS

- VueJS

- GAMES!

- Etc.

Photino vs. Electron

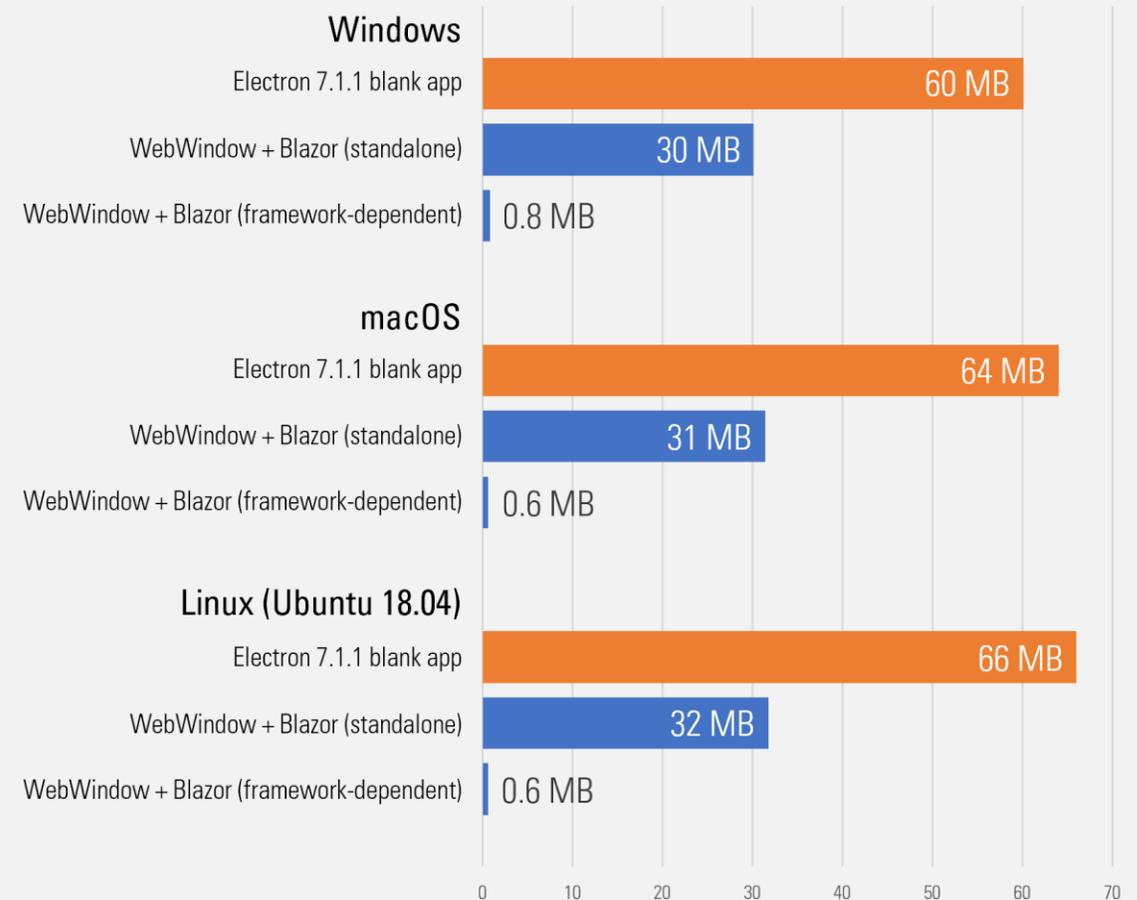**Photino**

- Photino does not require Node.js as the host application

- Photino is smaller and lighter weight because it doesn't require downloading and installing the Chromium engine or Node.js

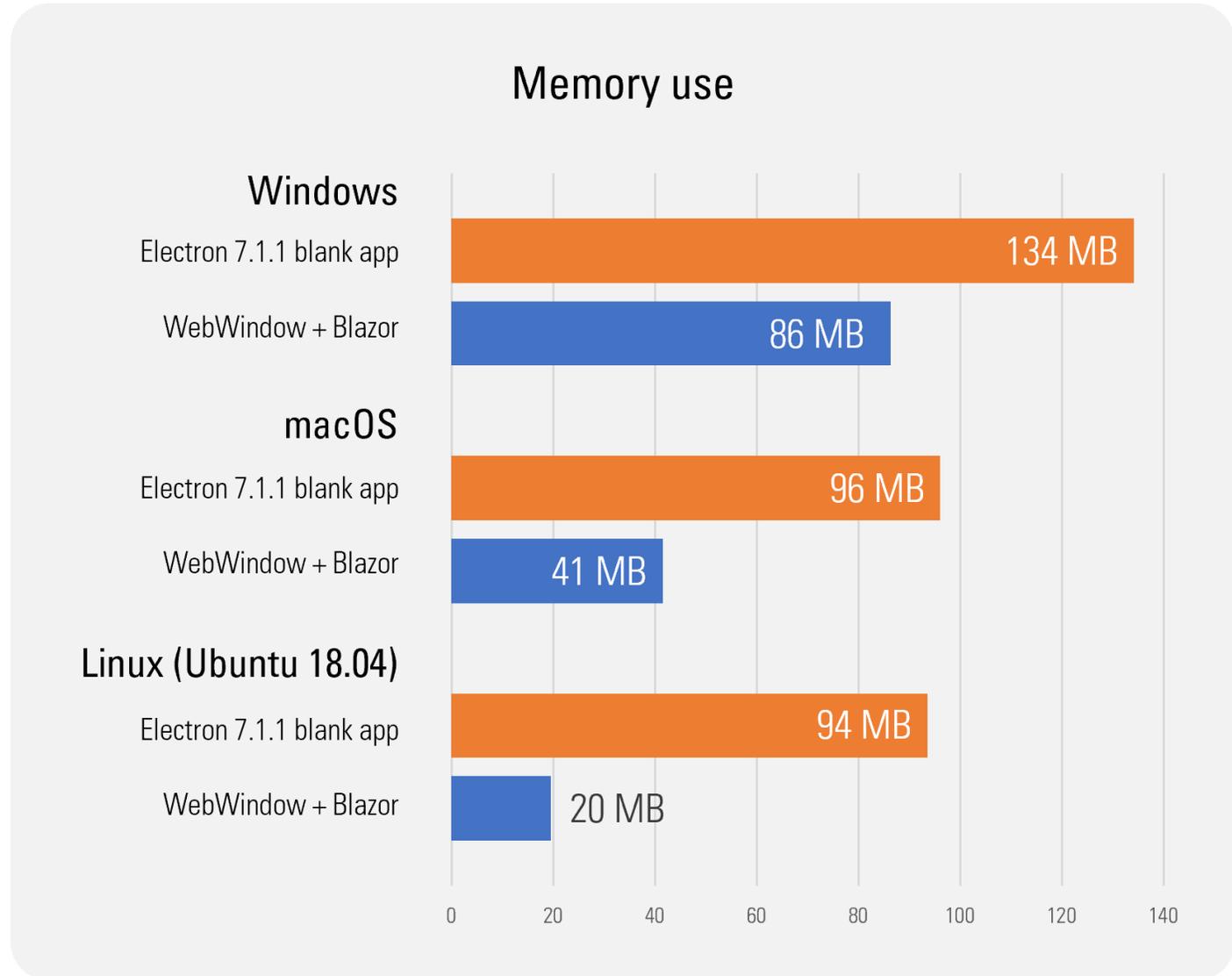- If .NET 5 is already installed on the OS, the difference is huge

Download size (compressed)

Windows

Electron 7.1.1 blank app	60 MB
WebWindow + Blazor (standalone)	30 MB
WebWindow + Blazor (framework-dependent)	0.8 MB

macOS

Electron 7.1.1 blank app	64 MB
WebWindow + Blazor (standalone)	31 MB
WebWindow + Blazor (framework-dependent)	0.6 MB

Linux (Ubuntu 18.04)

Electron 7.1.1 blank app	66 MB
WebWindow + Blazor (standalone)	32 MB
WebWindow + Blazor (framework-dependent)	0.6 MB

0   10   20   30   40   50   60   70

Photino vs. Electron

- Photino uses far less memory than Electron
- Especially on Mac and Linux where WKWebView and WebKitGTK+2 use far less memory than Chromium

Memory use

Windows	
Electron 7.1.1 blank app	134 MB
WebWindow + Blazor	86 MB
macOS	
Electron 7.1.1 blank app	96 MB
WebWindow + Blazor	41 MB
Linux (Ubuntu 18.04)	
Electron 7.1.1 blank app	94 MB
WebWindow + Blazor	20 MB

0 20 40 60 80 100 120 140

Photino

- tryphotino.io
- github.com/Tryphotino
- docs.tryphotino.io