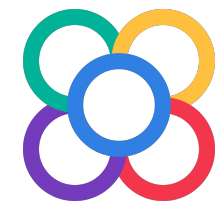


WASM CAN DO THAT!?



Daniel López

@vomkriege



About me

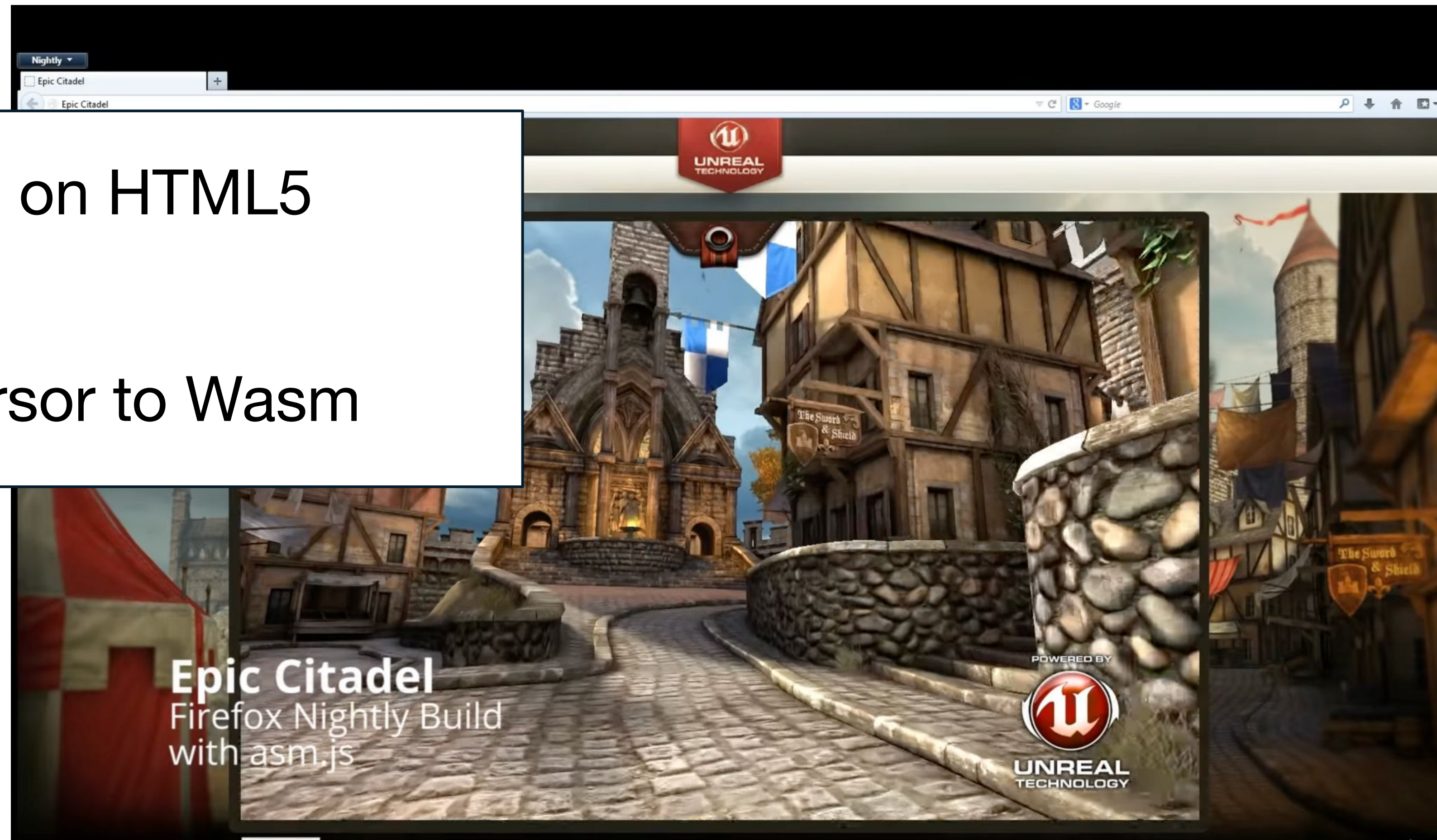
- Involved with Open Source since 1993
- Co-founder of Bitnami and Wasm Labs
- Focus on making OSS accessible to all



WHY WASM?

The demo that started it all

- Unreal Engine 3 running on HTML5
- 2013: asm.js, the precursor to Wasm



GAMES

- Portability across devices, in particular browser (incl WebGL support)
 - Unity, Reverie Engine (OSS), Unreal OSS web port, Rive.app
- Extensibility via Wasm: Flight Simulator
- Related talks: Wander (High Perf Graphics Wasm), Extism

ARTIFICIAL INTELLIGENCE

- Multiplatform deployment
 - Browser-based leveraging WebGPU
 - Standardization around WASI-nn
- AI Controller Interface: Prompts as Wasm programs (<https://github.com/microsoft/aici>)
- Related talks: WasmEdge Inference Workshop and LLM extensions , Pipeless, Spin ML

Modern websites in a Raspberry Pi Zero with WebAssembly

The screenshot shows a web browser displaying a website titled "The Modern Web Anywhere With WebAssembly". The website content includes:

- Header: "Wasm Labs @ VMware OCTO"
- Main heading: "The Modern Web Anywhere With WebAssembly"
- Text: "AN EXPERIMENT BY WASM LABS"
- Text: "You may think this looks like a regular site, and you are right. We wanted to create a site using modern tools and techniques, including Server Side Rendering (SSR) and run it in a particular environment."
- List of features:
 - It's based on modern frameworks and tools: [React](#), [Vite](#) and [MDX](#)
 - It prerenders the initial content using [Server Side Rendering \(SSR\)](#)
 - The SSR logic runs inside a WebAssembly module by embedding the [QuickJS](#) runtime (using [Javy](#))
 - The WebAssembly runtime we used is [Wazero](#)
 - It runs in a [Raspberry Pi Zero W \(BCM2835 / ARM 32-Bits 1GHz\)](#)
- Text: "Everything brought to you by the Wasm Labs @ VMware OCTO team."
- Section: "NAVIGATE THROUGH THE SITE"
- Text: "We created a set of sample pages:"
- List of links:
 - [About Us](#)
 - [Other projects](#)
- Footer: "Do you want to stay up to date with WebAssembly and our projects?"

On the right side of the screenshot, a terminal window shows the following output:

```
pi@raspberrypi:~$ lscpu
Architecture: armv6l
Byte Order: Little Endian
CPU(s): 1
On-line CPU(s) list: 0
Thread(s) per core: 1
Core(s) per socket: 1
Socket(s): 1
Vendor ID: ARM
Model: 7
Model name: ARM1176
Stepping: r0p7
CPU max MHz: 1000.0000
CPU min MHz: 700.0000
BogoMIPS: 697.95
Flags: half thumb fastmult vfp edsp java tls
pi@raspberrypi:~$ ./server ssr.wasm
Listening on 0.0.0.0:8080
"/" Render page from Wasm
```



Using WASM and WASI to run .NET 7 on a Raspberry Pi Zero 2 W

SMALL DEVICES

- Edge/IoT: From centralized and homogeneous Cloud to distributed and heterogeneous
- Wasm an unifying platform for disparate devices (ie. streaming apps)
- Small footprint means reach where containers and VMs can't
- Modern tooling to unexpected devices, removing vendor constraints
- Related talks: Programmable Embedded Vision Sensors, Wasm TinyGo

EMULATION

- From running in multiple platforms to ... Running Platforms in Wasm!
- From preserving video games to backwards compatibility (Flash, Applets) to instant development and learning sandbox environments
- WALI: The WebAssembly Linux interface
- Related talks: ELFConv

KUBERNETES

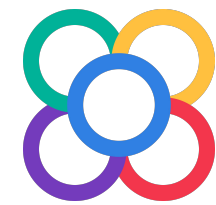
- Wasm and Kubernetes, better together?
- Wasm provides a lot of the benefits of Kubernetes without the cost and complexity
- Related talks: Production Wasm workshop

WHAT'S NEXT?

The best way to predict the future is to create it

Peter Drucker

THANKS!



Daniel López

@vomkriege

