

The roof is on fire?

* There might be loud sounds and flashing lights during this presentation.

* Slides theme by @antfu.

Hi, I'm Theodore.

Software Engineer.

Based in Athens.

Working at Proxima Analytics.

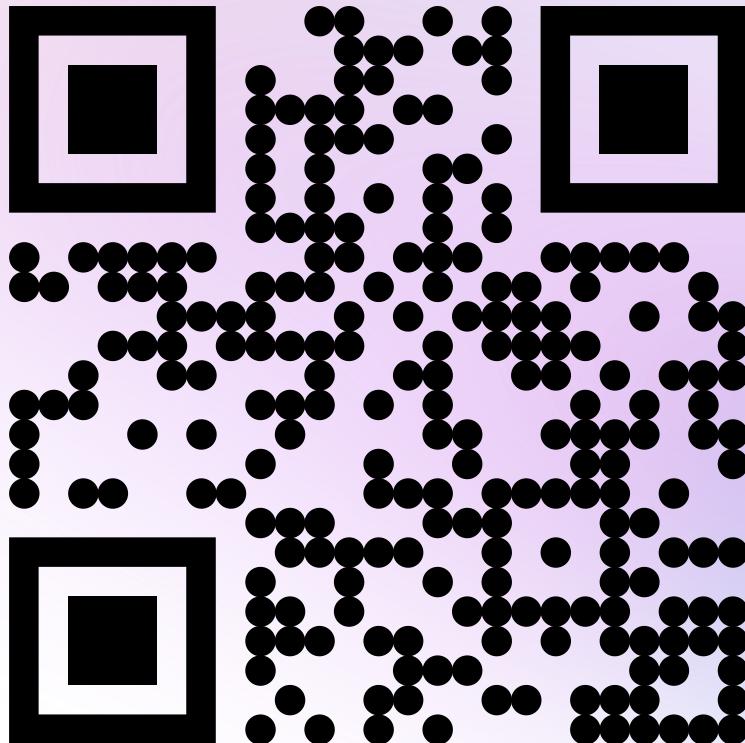
vorillaz.com [@vorillaz](https://github.com/vorillaz)

June, 2021

2022, 2023 and so on.



Photo by Konstantinos Tsakalidis / SOOC



pyRomania
<https://dub.sh/forests>

* Furniture, meetballs and illegal timbering.

How fires are "working"?

We **need** to understand their nature:

- ⌚ **Part of the ecosystem.** Mediterranean loves sparks.
- 🏃 **They are spreading quickly.** ~ 20km per hour.
- ▣ **They are unstoppable.** Once hitting a critical point.
- 🔥 **Their sources are unpredictable.** Cigarettes, campfires, fireworks, thunderstorms, farm chores etc.
- ☎ **Quick response is crucial.** Alert and deploy aerial or ground units.

OK, let's build a device.

A monitoring and alerting system for wildfires with a few goals in mind:

- Easy to deploy.
- Low cost.
- Accessible.
- Easy to build and maintain.
- Open-source. *Duh!*

Microcontrollers.

Arduino, Raspberry Pi, ESP32

Code?

C or C++

Python & Node.js.

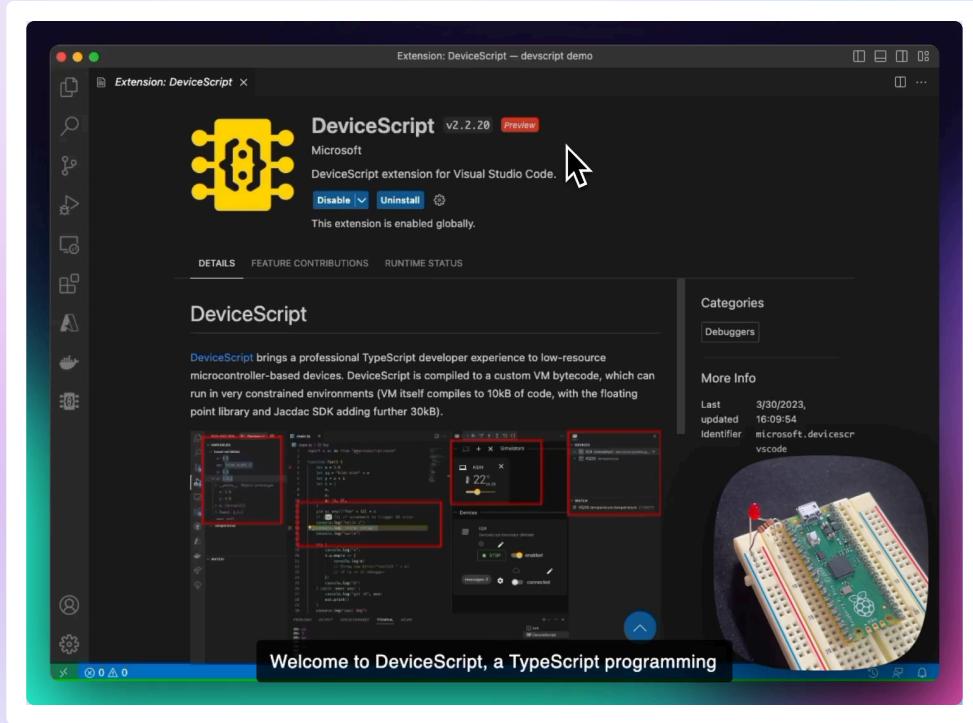
JavaScript?

It's a great language for building IoT devices.

- Event driven.
- Great registry of libraries.
- Mix and match with the browser, backend APIs, mobile apps and more.
- Lot's of options (Cylon.js, NodeBots, Johnny-Five, Espruino, etc.)

DeviceScript

- TypeScript.
- Great VSCode & Platform.io support.
- Target multiple devices.
- Core libraries & APIs.
- From TypeScript to bytecode.
- Simulation & debugging.



microsoft.github.io/devicescript

Electronics 101

A beautiful
live demo
lives here.

Hello world.

```
import { pins } from "@dsboard/esp32c3_bare";
import { startLightBulb, startButton } from "@devicescript/servers";

const led = startLightBulb({
  pin: pins.D3,
});

const button = startButton({
  pin: pins.D10,
});

button.down.subscribe(async () => {
  // toggle LED on/off
  console.log(`toggle`);
  await led.toggle();
});
```

Moving on.

Translating signals to voltage.

Temperature & Humidity.

```
import { startAHT20 } from "@devicescript/drivers";
import { schedule } from "@devicescript/runtime";

const { temperature, humidity } = await startAHT20();

schedule(
  async () => {
    const tmp = await temperature.reading.read();
    const hum = await humidity.reading.read();
    console.log(`Temperature: ${tmp}°C, Humidity: ${hum}%`);
  },
  { interval: 5000 },
);
```

Sensing.

{"waiting": "..."}

A beautiful
live demo
lives here.

Gas & Flame sensors.

```
import { startSensors } from "./detect";
import { schedule } from "@deviceScript/runtime";

const { flame, gas } = await startSensors();

const threshold = 10;

schedule(
  async () => {
    const flameOn = await flame.reading.read();
    const gasOn = await gas.reading.read();
    if (gasOn > threshold || gasOn) {
      console.log("🔥🔥🔥🔥");
    }
  },
  { interval: 5000 },
);
```

Fire in the hole.

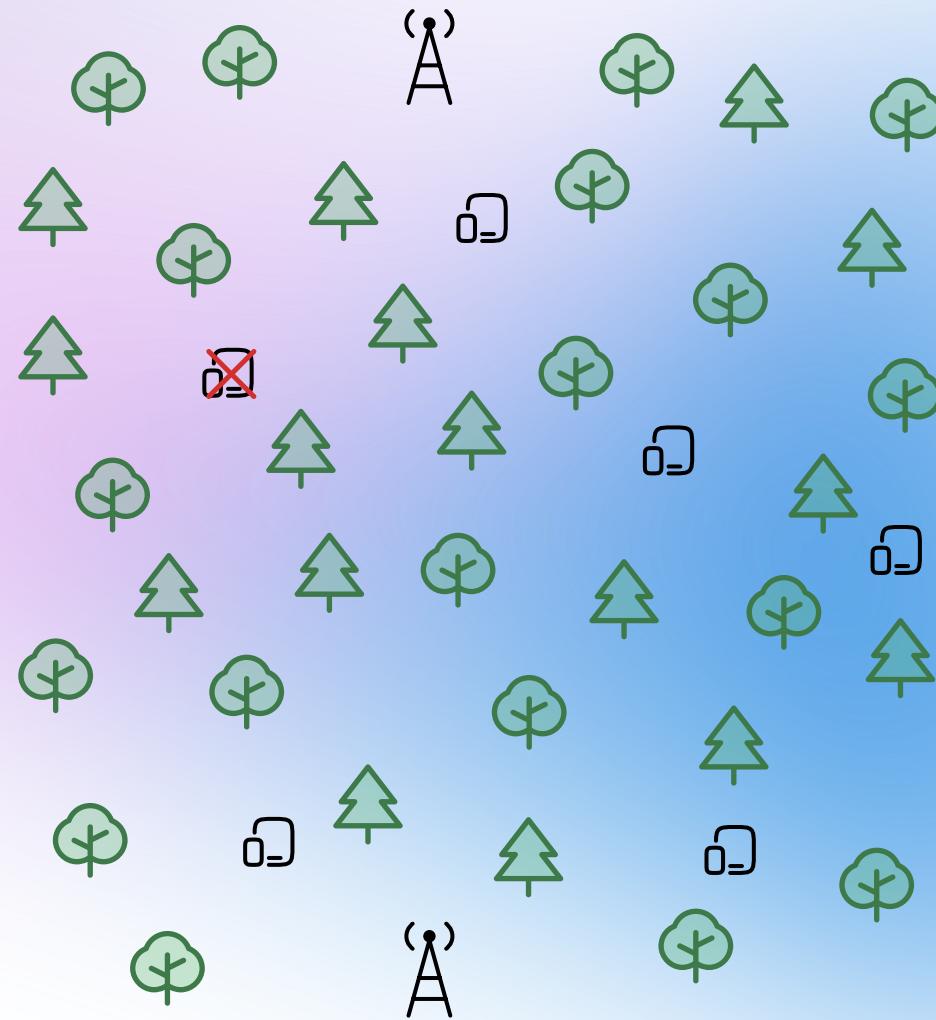
A beautiful
live demo
lives here.

Bleep, Blop, Bloop

* Working in clusters

Painless Mesh

npm.io/painlessmesh.ts



Bringing it all together.

From prototype to production.

- Solar powered and battery backed.
- Squeeze into a board.
- 3D printed case.
- CE certified.
- First demo cluster in the wild (March 2024).

But there is more.

A lot more.

Hello LoRa.

WAN clusters, up to 30km.
Off-grid, decentralized networks.

meshtastic.org

AI.

As all the cool kids do.

Prophet.

Failure forecasting as an industry standard.

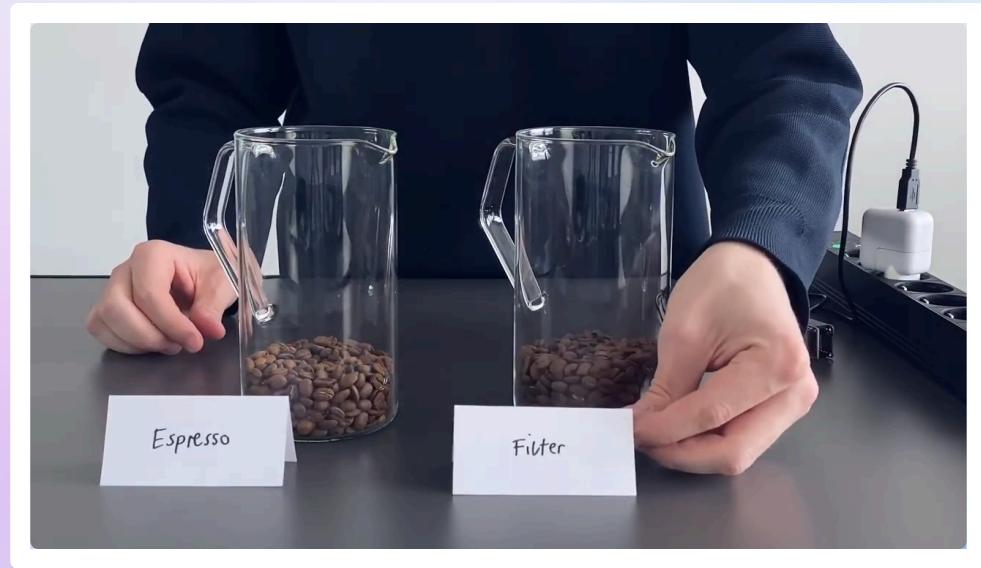
TinyML.

AI in microcontrollers

AI in the Sensor

BME688 by Bosch

- Train with sample dataset on the field.
- Better gas detection.
- Can detect smells and the lifecycle of the forest.



youtube.com/watch?v=4vdliMRtxBY

Yes And No

- We have the data (Copernicus, population density, Environmental)
- We need to correlate them somehow.

Can we forecast fires with AI?

With a different meaning though.

It's a wrap!

The roof is on fire?

Well, just let it burn.

Thanks a lot!

Slides & code on vorillaz.com