

TAMPERE UNIVERSITY OF TECHNOLOGY

NOKIA

On the Web Platform Cornucopia

Tommi Mikkonen, Cesare Pautasso, Kari Systä, Antero Taivalsaari

Abstract

The evolution of the Web browser has been organic, with new features introduced on a pragmatic basis rather than following a clear rational design. This evolution has resulted in a cornucopia of overlapping features and redundant choices for developing Web applications. These choices include multiple architecture and rendering models, different communication primitives and protocols, and a variety of local storage mechanisms. In this position paper we examine the underlying reasons for this historic evolution. We argue that without a sound engineering approach and some fundamental rethinking there will be a growing risk that the Web may no longer be a viable, open software platform in the long run.

Cornucopia



- Organic ecosystem evolution
- Overlapping/Redundant Capabilities
- Incompatible and incoherent abstractions
- Fashion-driven development
- "Cargo-cult" programming

Web as a Platform

Web App

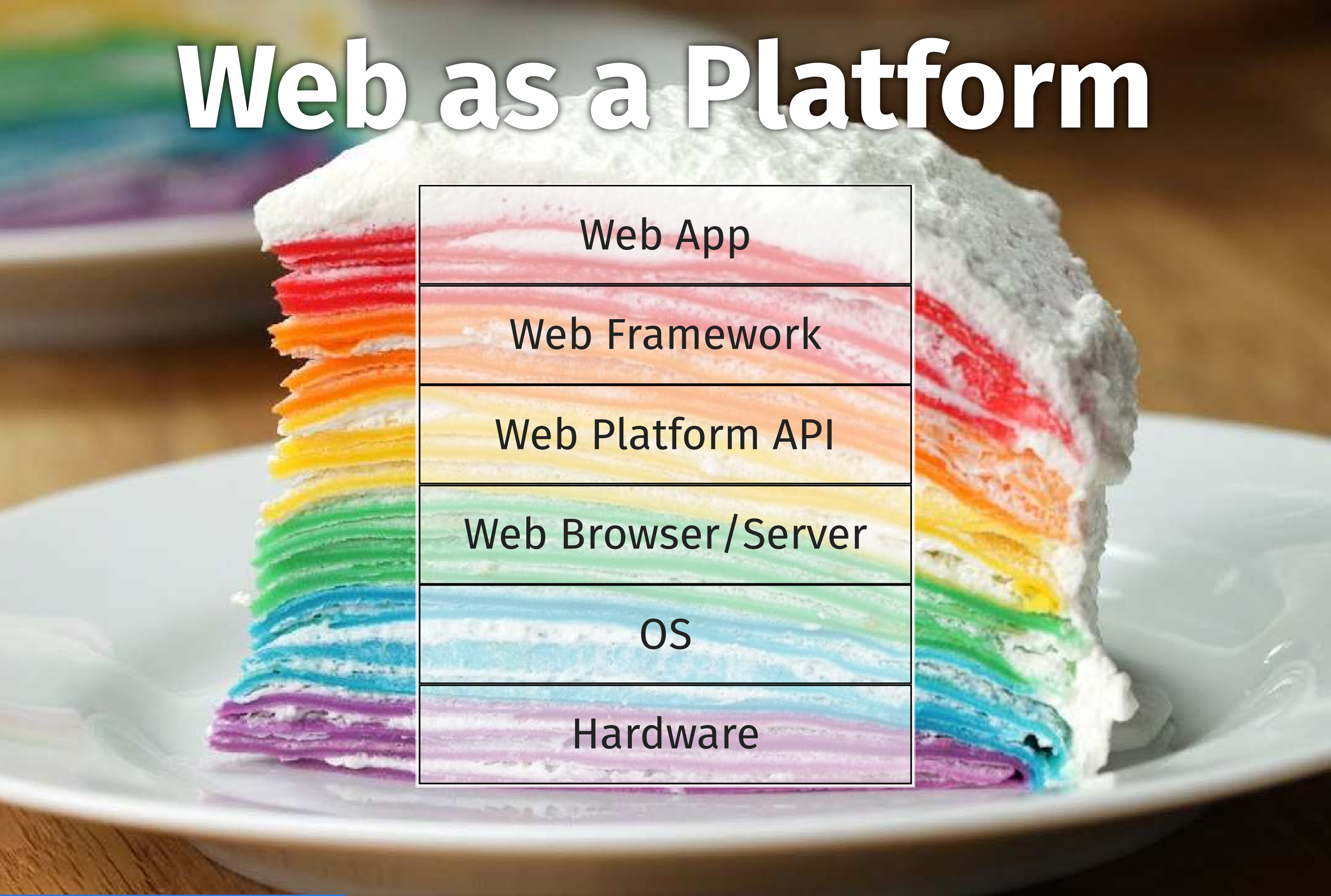
Web Framework

Web Platform API

Web Browser/Server

OS

Hardware



Web Platform APIs

144 HTML5 Specifications

Rendering Models

DOM, Canvas, SVG, WebGL, Web Components

Storage

Cookies, Local Storage, Indexed DB, File API, FileSystem API, (WebSQL)

Communication

XMLHttpRequest, Fetch, Web Sockets, WebRTC, Web Bluetooth, Web NFC

Multimedia

Web Audio, Web MIDI, Audio Output Devices, Audio Processing, Media Capture, Media Capture and Stream

Patterns for Web Platform API Formation



1. Vendor Features
2. Plugins
3. Hardware
4. Frameworks
5. Language Pre-compilers

Vendor Features

- Almost all the standard Web platform facilities we take for granted were once specific to one browser vendor

```
<IMG SRC="" >
```

Mosaic 1.0 (April 1993)

HTML 2.0, RFC1866, November 1995

```
-webkit- -o- -moz- -ms-
```

Vendor prefixes

Plugins

- Extensibility playground for adding features on the side

Flash Video (2002)

```
<object type="application/x-shockwave-flash"
  data="movie.swf"
  width="640" height="480">
  <param name="quality" value="high"/>
  <param name="wmode" value="window"/>
</object>
```

HTML5 Video (2007)

```
<video width="640" height="480" controls>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
</video>
```


Hardware

Direct standard access to new hardware capabilities or sensors

2011	Audio Processing
2013	Touch Events
2016	Battery Status, Geolocation
2017	Presentation, Encrypted Media Extensions (DRM), Audio Output Devices
2018	Media Capture, High Resolution Timer, WebGL (GPU), Web Audio
2019	Accelerometer, Ambient Light, Orientation, Gyroscope, Proximity, DeviceOrientation, WebXR

Frameworks

Abstractions of dominant frameworks gradually "leak down" into the standard platform.

```
$( '#id' ).html( "...");
```

jQuery (2006)

```
document.querySelector( '#id' )  
    .innerHTML = "...";
```

Safari 3.1 (March 2008)

[HTML5 Selector API \(W3C Recommendation February 2013\)](#)

Language Pre-Compilers

- Constructs of dominant language pre-compilers gradually shift into the standard JS/CSS languages.

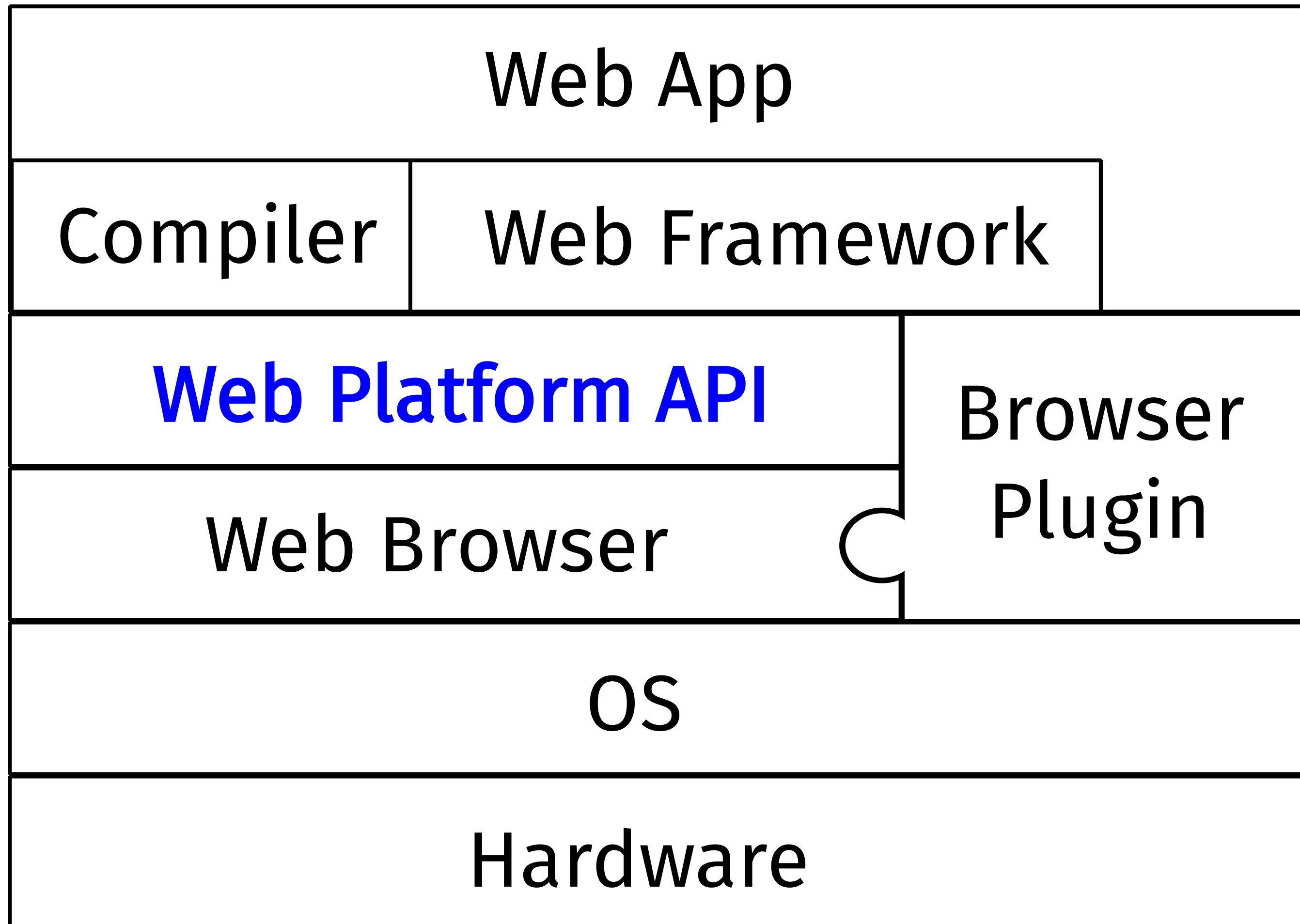
```
add = (a, b) => a+b
```

CoffeeScript (2010)

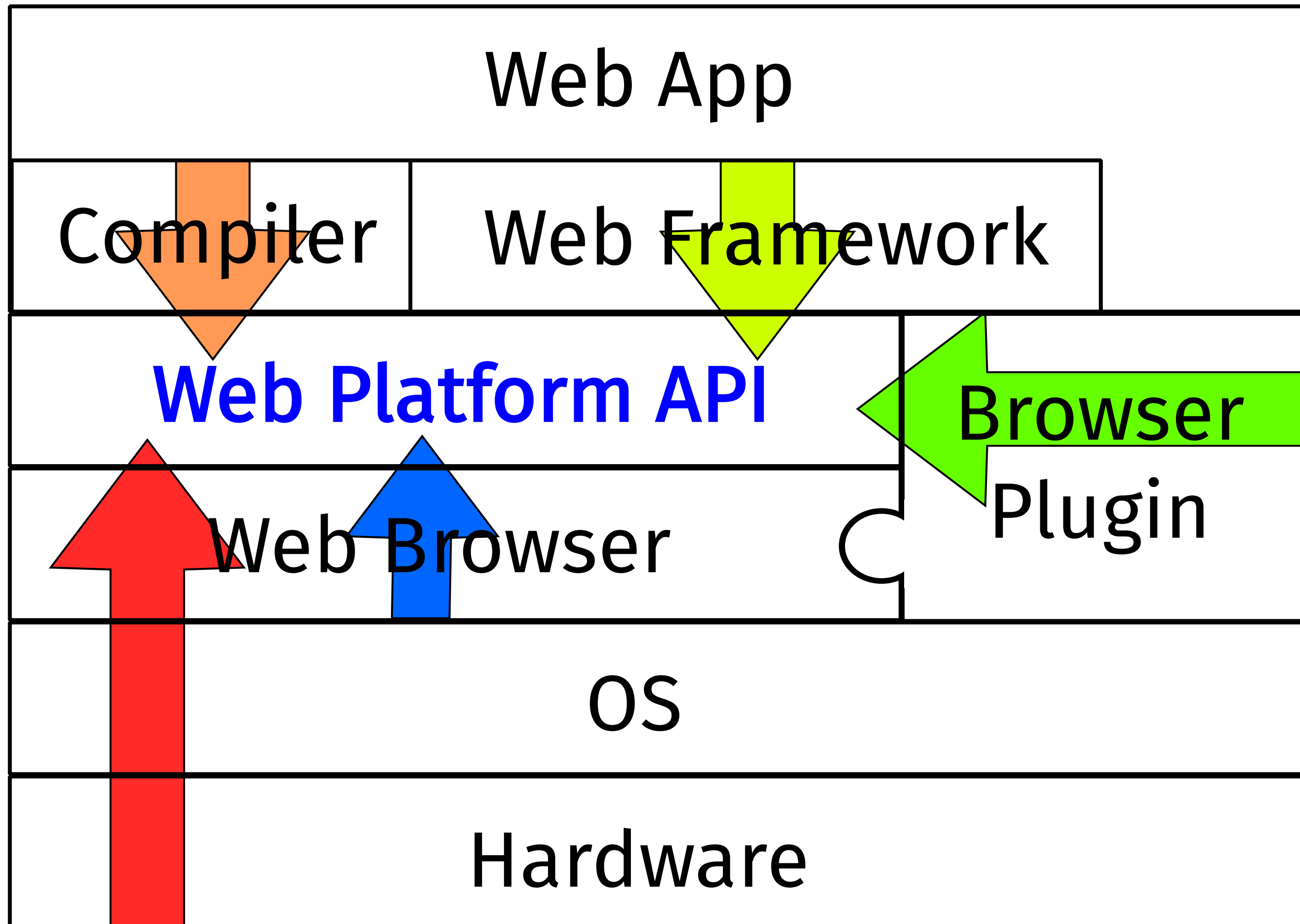
```
let add = (a, b) => { return a+b; }
```

JavaScript (ES2015)

Web Platform Architecture



Web Platform API Forces



**Is the
Web Platform Cornucopia
viable for the
Web ecosystem?**

(Some Open Questions)

Developer Impact

- What is the effort required to maintain a reasonably coherent Web platform?
- Are Web developers becoming more or less productive over the years?
- How much time and effort is spent by the developers in rewriting their code to follow the rapid evolution of frameworks, or to port Web applications between frameworks?

Long Term

- Is Web Engineering innovation dominated by frameworks or by browser API features (or vendors) underneath them?
- Will there be a coherent, long-lasting set of key frameworks that are deliberately set to drive browser API evolution?
- Will browser vendors give up, leading to a "**monoculture**" in which only one browser engine remains?
- While such a monoculture would have a better chance to keep the feature cornucopia under control, would it really be the desirable end state for the long term sustainability of the Web?

References

- Marc Andreessen, [proposed new tag: IMG](#), 25 February 1993
- Bruce Lawson, [CSS Vendor Prefixes in Opera 12.50 Snapshots](#), 10 August 2012
- Erik Wilde, [HTML5 Overview](#)
- Antero Taivalsaari, Tommi Mikkonen, Cesare Pautasso, Kari Systä, [Comparing the Built-In Application Architecture Models in the Web Browser](#), Proc. of the 1st IEEE International Conference on Software Architecture (ICSA 2017), Gothenburg, Sweden, April 2017