



Project Fugu

What we have enabled



Kenneth Rohde Christiansen (Intel)

Slides: goo.gle/project-fugu-tpac2022



Alex Russell & Diego González (Microsoft)



Ben Morss & Thomas Steiner (Google)



What is Project Fugu?



The Capabilities Project (code name Project Fugu) is a cross-company effort with the objective of making it possible for web apps to do anything platform-specific apps can.



intel®

 Microsoft

Google



How does Project Fugu operate?

Fixed process requiring explainer, spec, TAG review, security review,...

Try new ideas in origin trials first to avoid burn-in.

User agent optionality, e.g., for permission handling.

Spec \neq standard, but standardization is the ultimate goal.



All spec work starts in the WICG.



How does Project Fugu operate?

Standardization is happening:

Web Share API
W3C Candidate
Recommendation Draft



WebTransport
W3C Working Draft



File System
Living Standard



WebCodecs
W3C Working Draft



Clipboard API
and events
W3C Working Draft



Multi-Screen
Window Placement
W3C First Public Working Draft





How does Project Fugu operate?

Sometimes we also agree to disagree:

WebKit Feature Status

Search features...

Filters: Under Consideration, In Development, Partially Supported, Supported, Deprecated, Removed, Not Considering

13 features

- Battery Status API (Not Considering)
- HTML imports (Not Considering)
- Idle Detection (Not Considering)
- Magnetometer (Not Considering)
- Network Information (Not Considering)
- Proximity Sensor (Not Considering)
- Web Bluetooth (Not Considering)
- Web Bluetooth Scanning (Not Considering)
- Web MIDI (Not Considering)
- Web NFC (Not Considering)
- Web Serial (Not Considering)
- WebHID (Not Considering)
- WebUSB (Not Considering)

moz://a Specification Positions

This page tracks Mozilla's positions on open Web and Web-related specifications submitted to standards bodies like the IETF, W3C, and Ecma TC262.

Please remember, this isn't a commitment to implement or participate: **it's just what we think right now.** See [DevPlatform](#) to find out what we're implementing.

Want Mozilla's position on a specification? [Find out more.](#)

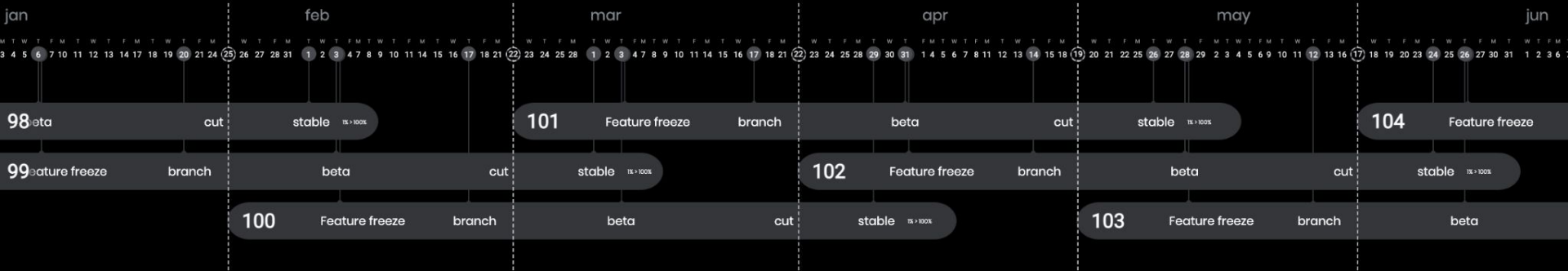
Expand All Collapse All

Showing 1 to 142 of 142 entries

Specification	Mozilla Position	More Info
File System Access	Support	More Info
File-Party Sets	Support	More Info
Get Installed Related Apps API	Support	More Info
ML Detection API	Support	More Info
Keyboard Map	Support	More Info
Low-level (PaaS) Sockets API	Support	More Info
Media Feeds	Support	More Info
Network Information API	Support	More Info
Periodic Background Synchronization	Support	More Info
Raw Clipboard Access	Support	More Info
Restricting Document Access of Same Origin Documents	Support	More Info
Serial API	Support	More Info
Signed HTTP Exchanges	Support	More Info
Web Background Synchronization	Support	More Info
Web Bluetooth	Support	More Info
Web NFC	Support	More Info
WebRTC API	Support	More Info
WebRTC Next Version Use Cases: Trickle JVB/SRT	Support	More Info
WebGL API	Support	More Info
Unlocks Emag QSD	Support	More Info
Generic Sensor API	Support	More Info
Orientation Sensor	Support	More Info
HTML imports	Support	More Info
JavaScript WebKit	Support	More Info

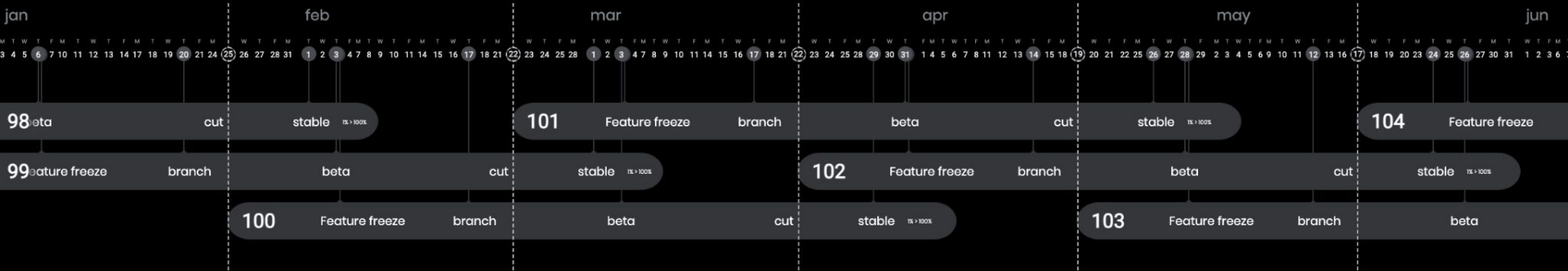


A walk down the Chromium release memory lane...





A walk down the Chromium release memory lane...





A walk down the Chromium release memory lane...

Vous êtes ici!



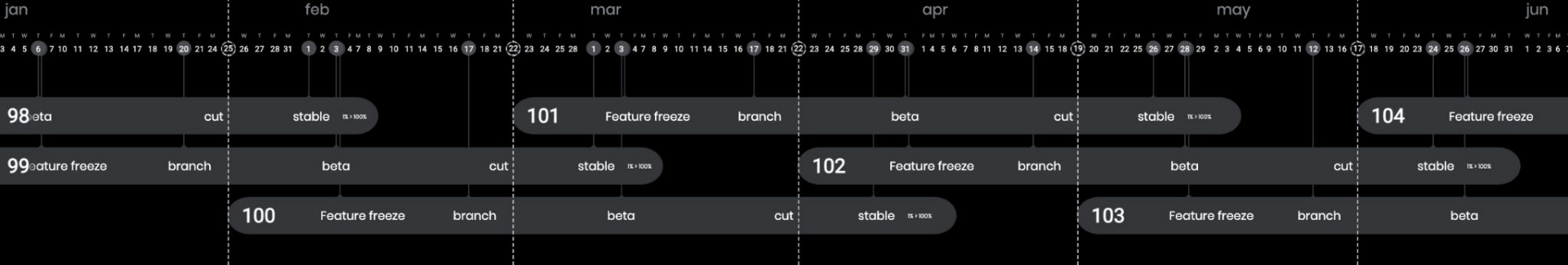
2022

TPAC

2021



TPAC





APIs introduced in...

94

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 Idle Detection



APIs introduced in...

94

Feature freeze

branch


beta

cut

stable

1% > 100%


👉 Idle Detection



Pause mobile notifications while you're using this device


To pause Chat mobile notifications while you're active on this device, allow your browser to detect if you're active or away. Click **Continue** and then **Allow** when prompted by your browser.

Continue No thanks



<https://mail.google.com/chat/>

mail.google.com wants to

 Know when you're actively using this device

Block Allow



APIs introduced in...

95

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 EyeDropper



APIs introduced in...

95

Feature freeze

branch

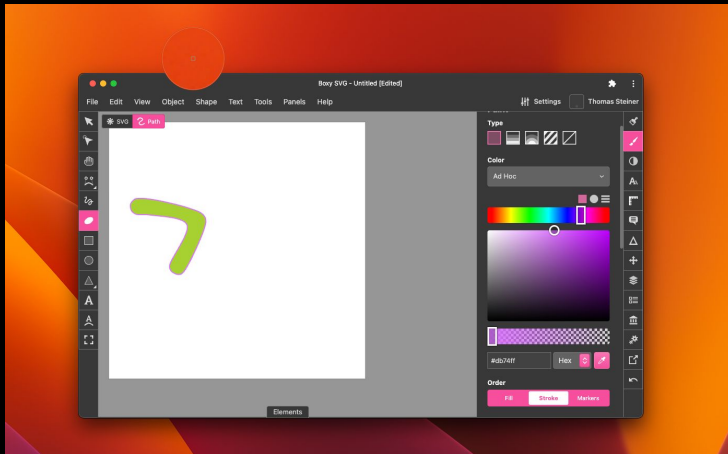
beta

cut

stable

1% > 100%

👉 EyeDropper



<https://boxy-svg.com/app>



APIs introduced in...

96

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 App Icon Shortcut menu for PWAs on macOS and Linux

👉 URL Protocol Handler Registration for PWAs



APIs introduced in...

96

Feature freeze

branch

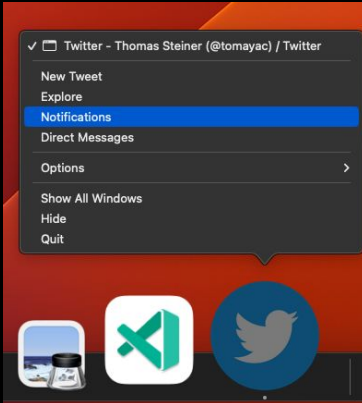
beta

cut

stable

1% > 100%

👉 App Shortcut menu for PWAs on macOS and Linux



<https://twitter.com/>



APIs introduced in...

96

Feature freeze

branch

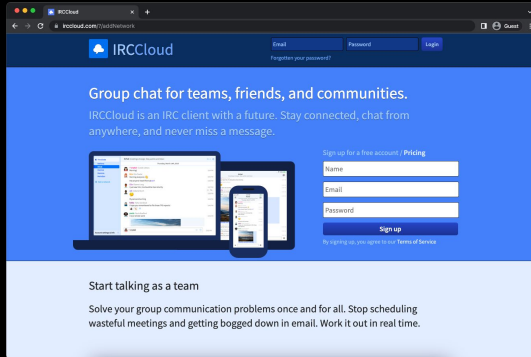
beta

cut

stable

1% > 100%

👉 URL Protocol Handler Registration for PWAs



<https://www.irccloud.com/>



APIs introduced in...

97

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 WebTransport



APIs introduced in...

97

Feature freeze

branch

beta

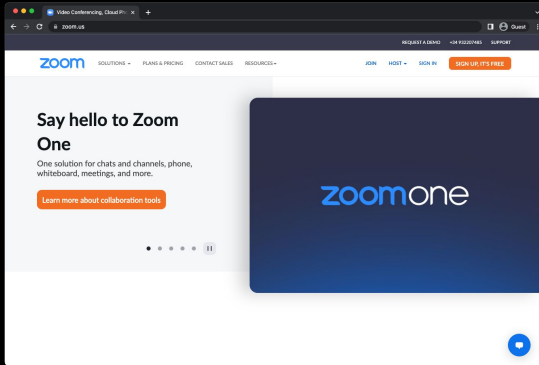
cut

stable

1% > 100%



WebTransport



<https://zoom.us/>



APIs introduced in...

99

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 PWA should be able to be uninstalled the same way a "real app" can



APIs introduced in...

99

Feature freeze

branch

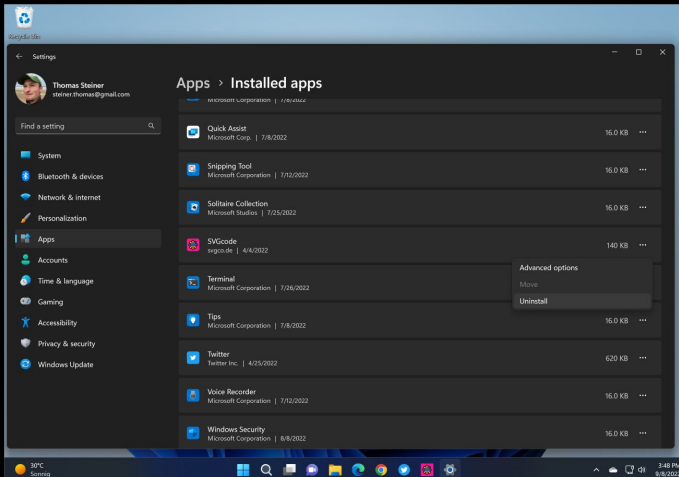
beta

cut

stable

1% > 100%

👉 PWA should be able to be uninstalled the same way a "real app" can



 Windows 11



APIs introduced in...

100

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 Web NFC: NDEFReader makeReadOnly()

👉 Multi-Screen Window Placement

👉 HIDDevice forget()



APIs introduced in...

100

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 Multi-Screen Window Placement

Multi-Screen Window Placement Demo ([GitHub](#), [ChromeStatus](#))

Open window URL: left: top: width: height:

Update screens Toggle fullscreen Open slide & notes Fullscreen slide Fullscreen slide and open notes

```
[0] "HP Z27n"
0.0 2560x1440
devicePixelRatio:1, colorDepth:24
isExtended:true, isInternal:false

[2] "HP Z27n"
2560.0 2560x1440
devicePixelRatio:1, colorDepth:24
isExtended:true, isInternal:false

window 1453,275 864x926

[1] "Built-in display" (Primary)
1444,1440 2048x1152
devicePixelRatio:1, colorDepth:24
isExtended:true, isInternal:true
```

<https://michaelwasserman.github.io/window-placement-demo/>



APIs introduced in...

101

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 USBDevice forget()

👉 Web USB sameObject behavior



APIs introduced in...

102

Feature freeze

branch

beta

cut

stable

1% > 100%

- 👉 **Window Controls Overlay for Installed Desktop Web Apps**
- 👉 **File Handling**
- 👉 **chrome.management API can no longer interact with PWA apps**



APIs introduced in...

102

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 Window Controls Overlay for Installed Desktop Web Apps

SQLite Viewer Web App

SQLite Viewer Web is a free, web-based SQLite Explorer, inspired by DB Browser for SQLite and Airtable. Use this web-based SQLite Tool to quickly and easily inspect .sqlite files. Your data stays private: Everything is done client-side and never leaves your browser.

Chinook_Sqlite.sqlite ▶ Album

Albumid	Title	Artistid
1	For Those About T...	1
2	Balls to the Wall	2
3	Restless and Wild	2
4	Let There Be Rock	1
5	Big Ones	3
6	Jagged Little Pill	4
7	Facelift	5
8	Warner 25 Anos	6
9	Plays Metallica By ...	7
10	Audioslave	8
11	Out Of Exile	8
12	BackBeat Soundtr...	9
13	The Best Of Billy C...	10



<https://sqliteviewer.app/>



APIs introduced in...

102

Feature freeze

branch

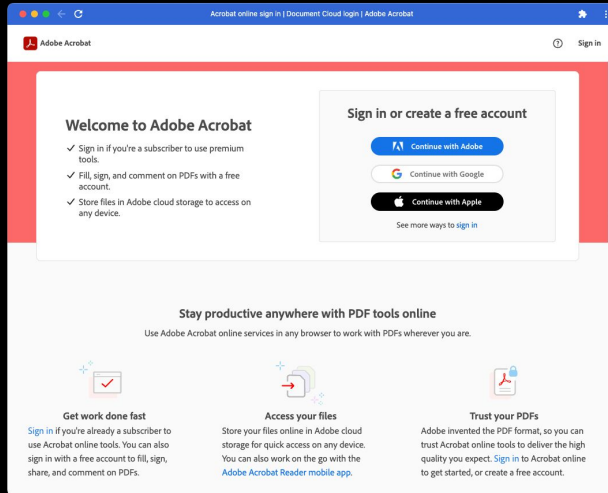
beta

cut

stable

1% > 100%

👉 File Handling



<https://documents.adobe.com/>



APIs introduced in...

103

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 SerialPort forget()

👉 Local Font Access



APIs introduced in...

103

Feature freeze

branch

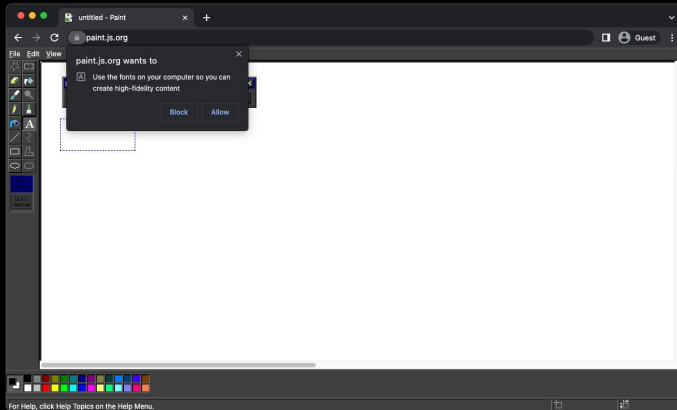
beta

cut

stable

1% > 100%

👉 Local Font Access



<https://paint.js.org/>



APIs introduced in...

104

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 Web Custom formats for Async Clipboard API



APIs introduced in...

102

Feature freeze

branch

beta

cut

stable

1% > 100%

👉 **Origin Private File System extension: AccessHandle**



APIs introduced in...

102

Feature freeze

branch

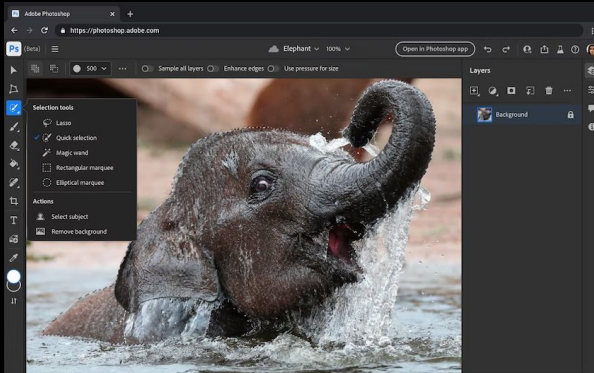
beta

cut

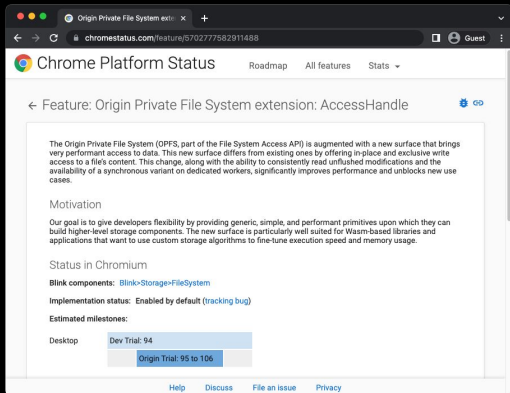
stable

1% > 100%

👉 Origin Private File System extension: AccessHandle



<https://photoshop.adobe.com/>



Chrome Platform Status

Feature: Origin Private File System extension: AccessHandle

The Origin Private File System (OPFS, part of the File System Access API) is augmented with a new surface that brings very performant access to data. This new surface differs from existing ones by offering in-place and exclusive write access to a file's content. This change, along with the ability to consistently read unflushed modifications and the availability of a synchronous variant on dedicated workers, significantly improves performance and unblocks new use cases.

Motivation

Our goal is to give developers flexibility by providing generic, simple, and performant primitives upon which they can build higher-level storage components. The new surface is particularly well-suited for WebAssembly-based libraries and applications that want to use custom storage algorithms to fine-tune execution speed and memory usage.

Status in Chromium

Blink components: [Blink-Storage+FileSystem](#)

Implementation status: Enabled by default (tracking bug)

Estimated milestones:

Platform	Dev Trial	Origin Trial
Desktop	94	95 to 106

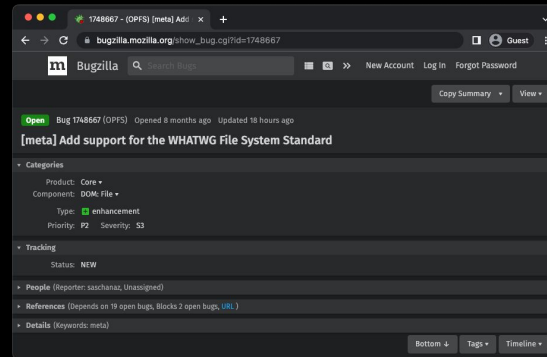
[Help](#) [Discuss](#) [File an issue](#) [Privacy](#)



The File System Access API with Origin Private File System

Feb 14, 2022 by Sihui Liu

It is very common for an application to interact with local files. For example, a general workflow is opening a file, making some changes, and saving the file. For web apps, this might be hard to implement. It is possible to simulate the file operations using IndexedDB API, an HTML input element with the `file` type, an HTML anchor element with the `download` attribute, etc, but that would require a good understanding of these standards and careful design for a good user experience. Also, the performance may not be satisfactory for frequent operations and large files.



1748667 - (OPFS) [meta] Add

bugzilla.mozilla.org/show_bug.cgi?id=1748667

Open Bug 1748667 (OPFS) Opened 8 months ago · Updated 18 hours ago

[meta] Add support for the WHATWG File System Standard

Categories

- Product: Core
- Component: DOM: File
- Type: enhancement
- Priority: P2
- Severity: S3

Tracking

Status: NEW

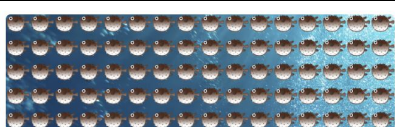
People (Reporter: sashchanz, Unassigned)

References (Depends on 19 open bugs, Blocks 2 open bugs, URL)

Details (Keywords: meta)

Bottom 4 | Tags | Timeline





Project Fugu API showcase

The Project Fugu API Showcase is a collection of apps that make use of APIs that were conceived in the context of Project Fugu.

Published on Monday, April 25, 2022 · Updated on Wednesday, June 1, 2022

Thomas Steiner
Tom is a Developer Advocate

Motivation

The Capabilities Project (code name [Project Fugu](#)) is a cross-company effort with the objective of making it possible for web apps to do anything platform-specific apps can. We enable amazing web applications like [PhotoLab](#) by exposing the capabilities of the underlying operating system platform to the web platform, while maintaining user security, privacy, trust, and other core tenets of the web.

But what are examples of some of the apps that make use of these capabilities? The Project Fugu API Showcase (embedded below) is sourced by community submissions and contains a filterable list of apps that make use of one or more APIs that were developed in the context of the project. You can propose missing apps by submitting them via an [accounting form](#). Submissions are reviewed on a regular basis and the showcase will be updated accordingly.

You can launch each app by clicking the app's name, the app's preview image, or the **Launch** app link. For many apps, you can also see the source code by clicking **Source code**. On supporting browsers, you can share any of the listed apps via the **Share** app feature. As a special incentive Easter egg, the Project Fugu API Showcase is of course [connected to the Project Fugu API Showcase](#). Happy browsing!

Showcase

Filter used APIs:

[Absolute Orientation Sensor](#) [Add to Home Screen](#) [App Clipboards](#)
[App Clipboards \(iOS/mac\)](#) [Background Fetch](#) [Background Sync](#) [Background](#)

[Cache Storage](#) [Compression Streams](#) [Content Ficker](#) [Content Index](#)

[Credential Lia Client](#) [Device Memory](#) [Digital Goods](#) [E-InkScreen](#) [File Handling](#)

[File System Access](#) [FileShare](#) [FullscreenOverlay](#) [Gamepad](#) [GamepadVibration](#)

[Launch Handler](#) [Local Acceleration Sensors](#) [Local File Access](#) [Media Capabilities](#)

[Media Session](#) [Navigation Previews](#) [Notification Triggers](#) [Payment Request](#)

[Periodic Background Sync](#) [Persistent Storage](#) [Pointer Lock \(Unstable/Movement\)](#)

[Protocol Handlers](#) [Push](#) [Relative Orientation Sensor](#) [Screen Wake Lock](#) [Service Worker](#)

[Stage Selection \(Unstable\)](#) [Storage](#) [Storage Exemptions](#) [WebGL Application White](#)

[Web Analytics](#) [WebAuthn](#) [WebBluetooth](#) [WebGPU](#) [WebPage Embedding](#) [WebShare \(iOS\)](#)

[Web Share \(iOS\)](#) [WebShare \(macOS\)](#) [WebShare \(Windows\)](#) [WebShare \(Android\)](#) [WebShare \(Linux\)](#)

[WebShare \(Android\)](#) [WebShare \(Linux\)](#) [WebShare \(macOS\)](#) [WebShare \(Windows\)](#)

Search apps:

[md.edf]

Tuner | listen to radio, podcasts and create playlists

Listen to radio, podcasts and create playlists.

[Launch app](#)
[Source code](#)

Used APIs:

[Add to Home Screen](#) [File Handling](#)
[File System Access](#) [Launch Handler](#)
[Local File Access](#)

Used APIs:

[Media Session](#) [Service Worker](#)
[Web Share](#)

For more examples, check out the Project Fugu API Showcase



What are we building?



Talking to developers

~~Oo, that looks shiny~~

What to build?

What's happening on
the web now?

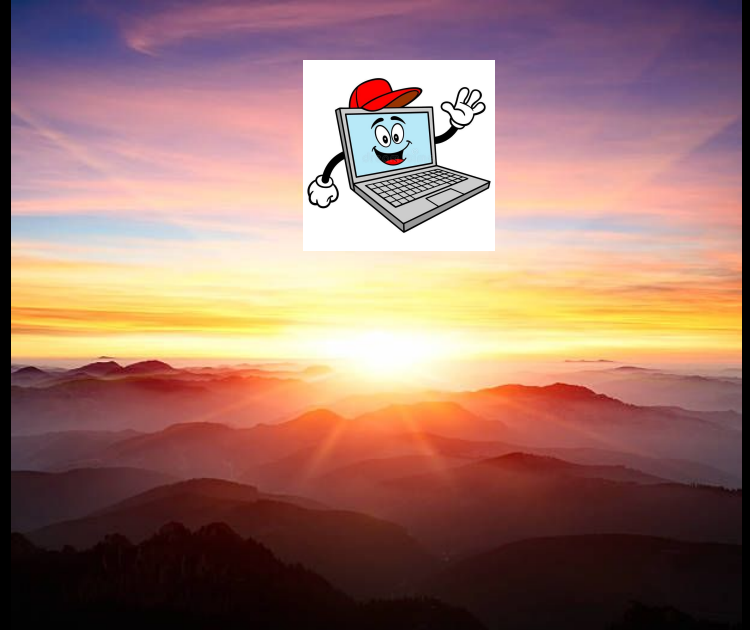
Help that happen.
Make it better.



MVP → v 2.0

New web use cases

- Video and audio creation apps
- Game streaming
- Things being aware of nearby things
- Educational toys like LEGO™
- Code editors
- Peripheral configuration
- Medical applications
- Call control
- Creative applications
- Retail...?



New capabilities

- Multiple screens
- Screen refresh rate
- BLE scanning, Serial over Bluetooth
- Better advanced storage
- File system access
- Low-latency audio





Project Fugu

Thank you!



Kenneth Rohde Christiansen (Intel)

Slides: goo.gle/project-fugu-tpac2022



Alex Russell & Diego González (Microsoft)



Ben Morss & Thomas Steiner (Google)