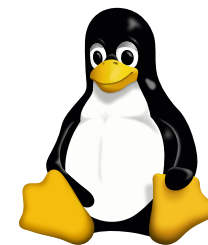


The Web Platform on Linux devices with WebKit: where are we now?



Mario Sánchez-Prada

mario@igalia.com

FOSDEM 2026

Brussels, January 31st 2026

About Me

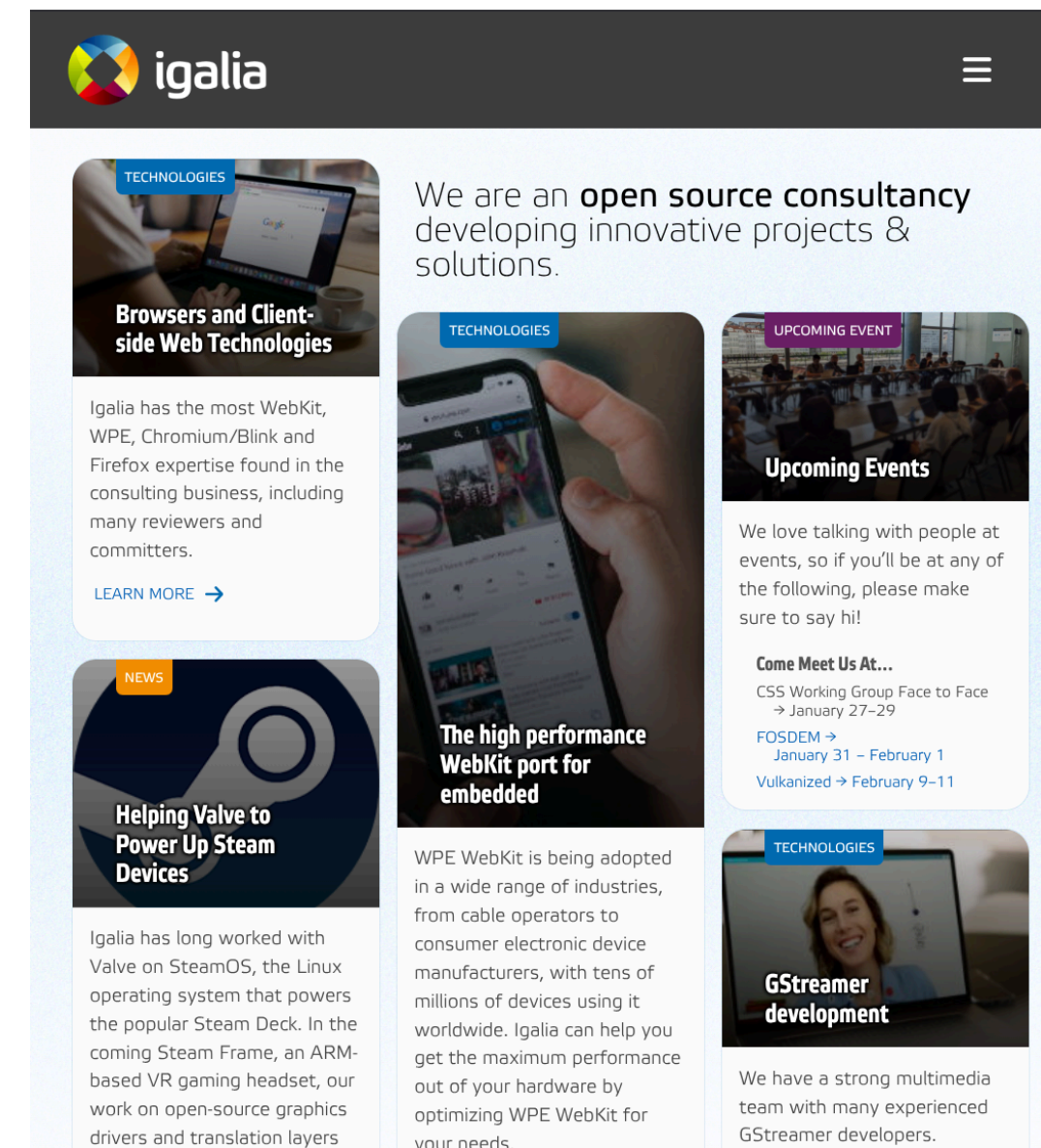
- **Software Engineer** and partner at **Igalia**.
- **Open Source work**: GNOME, Chromium/Blink, WebKit.
- **Other**: Maemo, Endless OS, Samsung Smart TV.
- Currently **coordinating** Igalia's **WebKit team**.



About Igalia

- **Founded in A Coruña (Spain)** in 2001.
- Specialized **Open Source consultancy**.
- **Fully remote** and with a **flat structure**.
- Second-largest **contributor** to the main Open Source **Web Rendering Engines**.
- **Other OSS work**: kernel, compilers, multimedia, graphics, drivers...
- Members of different **Working groups**:
 - W3C, WHATWG, TC39, Test262, Khronos...

<https://www.igalia.com>



Outline

1. Web Rendering Engines
2. What is WebKit?
3. WebKitGTK & WPE WebKit
4. History of WebKitGTK & WPE WebKit
5. Latest updates
6. Next steps

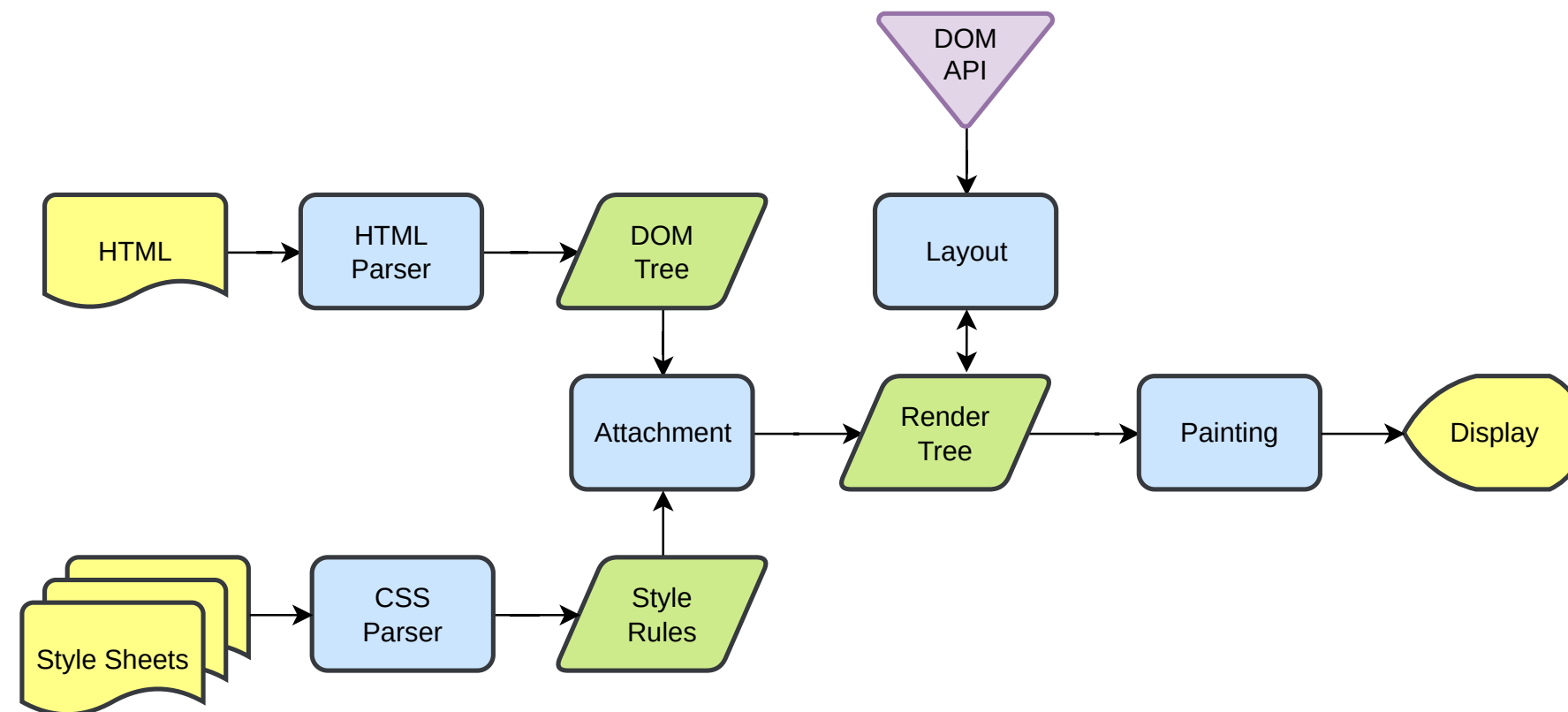


Web Rendering Engines



What is a Web rendering engine?

- **Fetches** HTML / CSS / JavaScript content from multiple sources.
- **Interprets** the web content to create an internal representation.
- **Renders a result** that users can **interact with**.



Main Web rendering engines



What is WebKit?



<https://webkit.org>



What is WebKit?

- Open Source **Web rendering engine** since 2005.
 - Forked from KHTML and KJS by Apple in 2001.
 - Forked again by Google in 2013 (*Blink*).
- **Main goals:**
 - Performance, portability, stability, compatibility, standards compliance, security, hackability and *embeddability*.
- Support for **different platforms:**
 - **Desktop & Mobile:** Mac, iOS, Linux, Windows.
 - **Embedded:** set-top-boxes, gaming consoles, smart home appliances, IVI systems, GPS navigation, digital signage...
- **Multi-process** architecture:
 - UI Process, Web Process, Network Process, GPU Process...



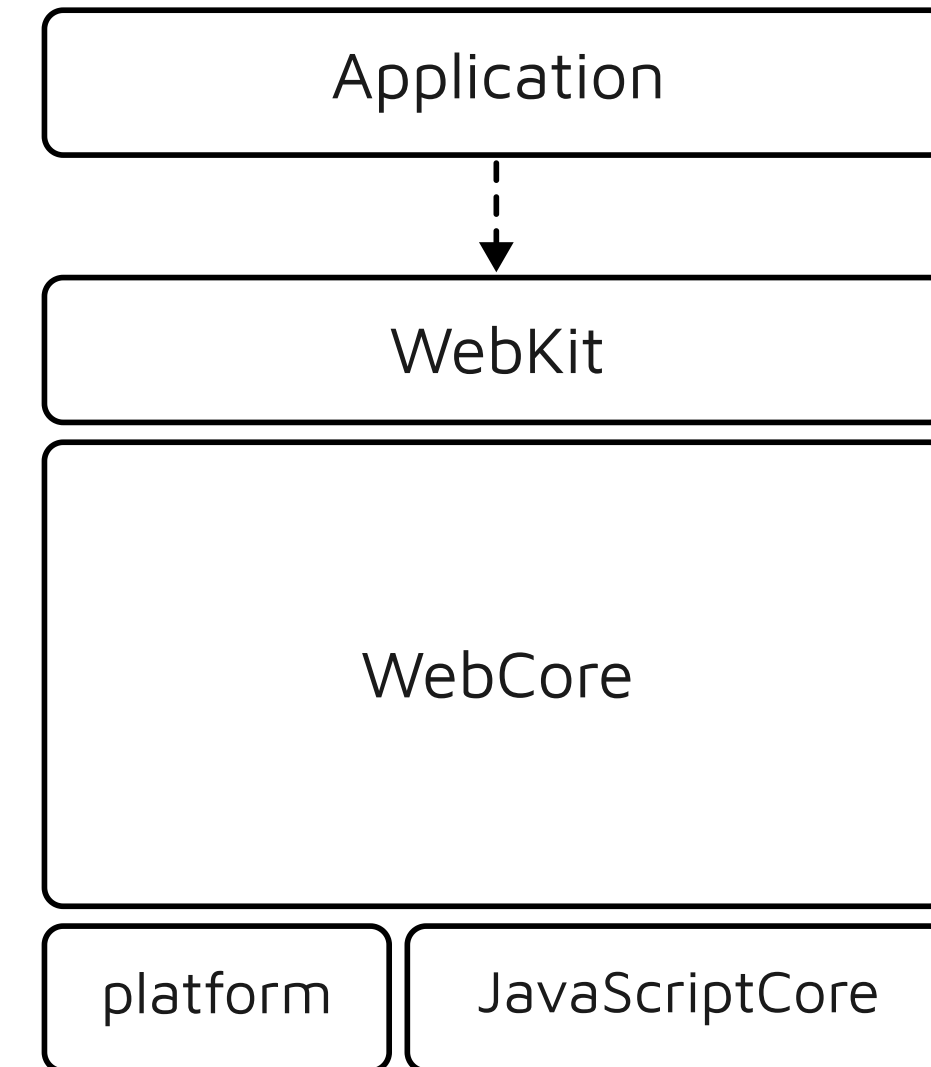
Advantages of WebKit

- 🕸 **Complete implementation** of the Web Platform
- 🔌 **Embeddable** as top priority
- 🧩 **Flexible and modular** architecture
- 🚀 **Performance** and **stability**
- 🔒 **Privacy** and **security** by design
- 🐧 **Independent** Linux-based flavours
 - 👉 Not controlled by any big corporation



WebKit architecture

- **Application:**
 - What the end-users interact with.
- **WebKit:**
 - Exposes a public API to applications and implements the multi-process model.
- **WebCore:**
 - HTML/CSS parsing, rendering, layout, painting, network, multimedia, accessibility...
- **JavaScriptCore:**
 - JavaScript engine (also supports WebAssembly).
- **Platform:**
 - Platform-specific hooks and low-level plumbing.

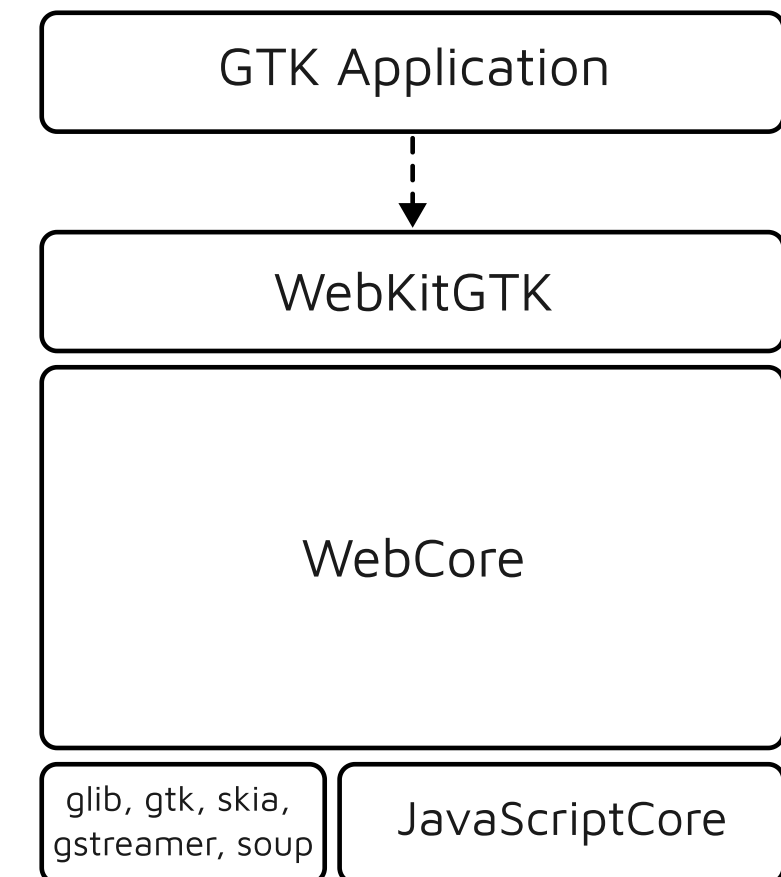


WebKit ports

WebKit port: adaptation of WebKit to a platform.

Official WebKit Ports (*upstream* ports):

- **Mac:** Safari, Apple Mail, iTunes, App Store...
- **iOS:** Web browsers on iOS devices (also Chrome).
- **Windows:** Microsoft Playwright, PlayStation SDK.
- **PlayStation:** PlayStation 4 & PlayStation 5.
- **WebKitGTK:** GNOME Web, Evolution, Shotwell...
- **WPE WebKit:** Custom "browsers" for embedded devices.



e.g. WebKitGTK port

<https://docs.webkit.org/Ports/Introduction.html>



WebKitGTK & WPE WebKit



What is WebKitGTK?

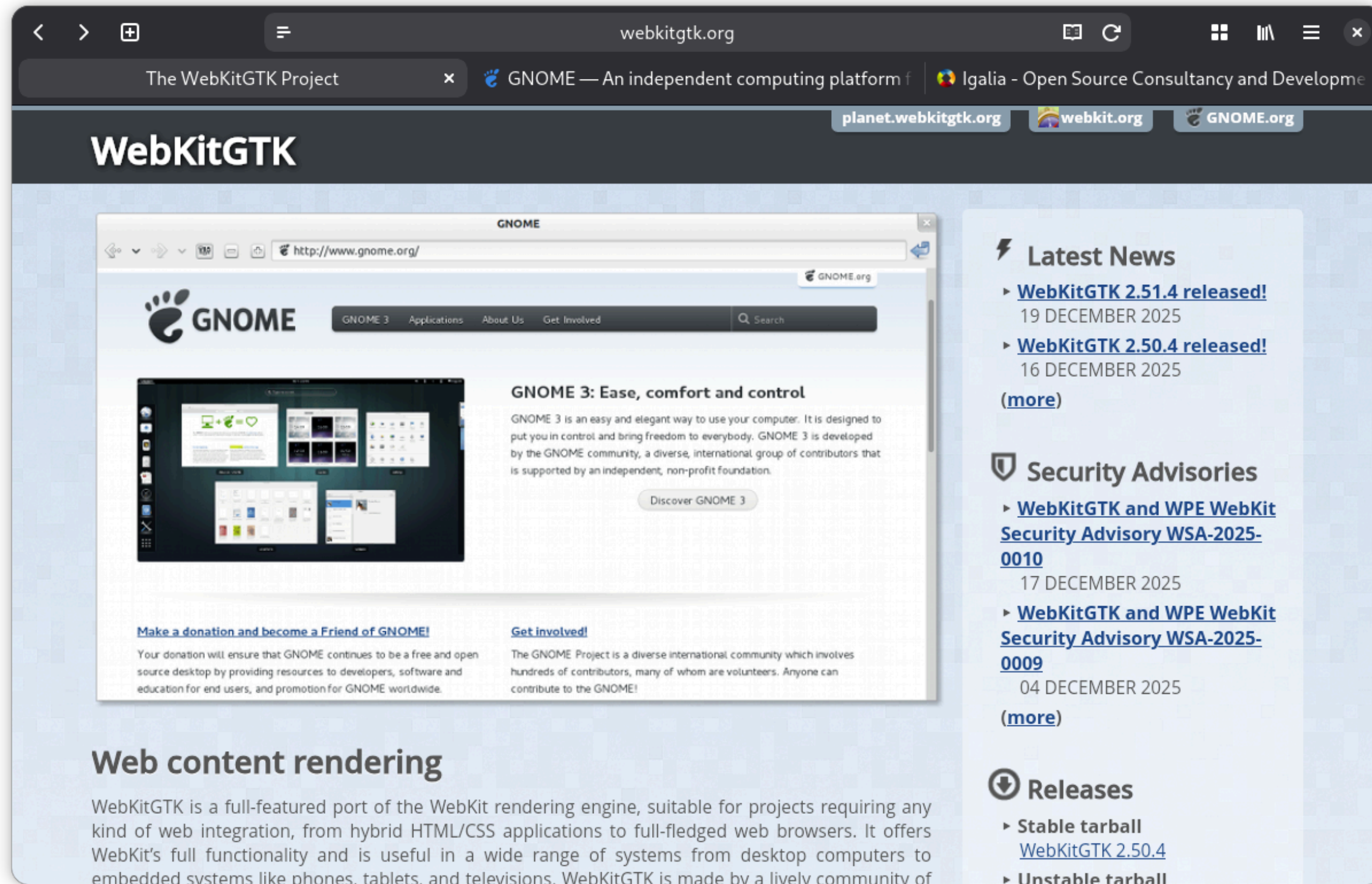
WebKitGTK: port of **WebKit** for Linux-based **GTK applications**.

- **Fully-featured** implementation of the **Web Platform**
 - Support for **HW-accelerated graphics** and **multimedia**.
- **Used** in a wide variety of **GTK applications**:
 - GNOME Web (browser), Evolution (mail client), GNOME Builder (IDE)...
- **Provides a *widget*** to be used in GTK applications: **WebKitWebView**
 - Offers a public API to interact with the Web engine and respond to events.
 - Platform-specific functionality is implemented on top of other components (e.g. Skia, OpenGL, EGL, GStreamer, libsoup, fontconfig, harfbuzz...).
- Fully operational **JavaScript engine** (JavaScriptCore)

 <https://webkitgtk.org>



What is WebKitGTK?



The screenshot shows a web browser window displaying the WebKitGTK project page. The browser's address bar shows 'webkitgtk.org'. The page features a dark header with the 'WebKitGTK' logo. Below the header, there's a section titled 'GNOME' which includes a screenshot of the GNOME 3 desktop environment. To the right of this, there's a 'Latest News' section with two entries: 'WebKitGTK 2.51.4 released!' dated 19 DECEMBER 2025 and 'WebKitGTK 2.50.4 released!' dated 16 DECEMBER 2025. Below the news, there's a 'Security Advisories' section with two entries: 'WebKitGTK and WPE WebKit Security Advisory WSA-2025-0010' dated 17 DECEMBER 2025 and 'WebKitGTK and WPE WebKit Security Advisory WSA-2025-0009' dated 04 DECEMBER 2025. At the bottom, there's a 'Releases' section with two entries: 'Stable tarball WebKitGTK 2.50.4' and 'Unstable tarball'. The page also includes a 'Web content rendering' section at the bottom left, which describes WebKitGTK as a full-featured port of the WebKit rendering engine.

WebKitGTK

GNOME

GNOME 3: Ease, comfort and control

GNOME 3 is an easy and elegant way to use your computer. It is designed to put you in control and bring freedom to everybody. GNOME 3 is developed by the GNOME community, a diverse, international group of contributors that is supported by an independent, non-profit foundation.

[Discover GNOME 3](#)

[Make a donation and become a Friend of GNOME!](#)

Your donation will ensure that GNOME continues to be a free and open source desktop by providing resources to developers, software and education for end users, and promotion for GNOME worldwide.

[Get involved!](#)

The GNOME Project is a diverse international community which involves hundreds of contributors, many of whom are volunteers. Anyone can contribute to the GNOME!

Web content rendering

WebKitGTK is a full-featured port of the WebKit rendering engine, suitable for projects requiring any kind of web integration, from hybrid HTML/CSS applications to full-fledged web browsers. It offers WebKit's full functionality and is useful in a wide range of systems from desktop computers to embedded systems like phones, tablets, and televisions. WebKitGTK is made by a lively community of

Latest News

- ▶ [WebKitGTK 2.51.4 released!](#)
19 DECEMBER 2025
- ▶ [WebKitGTK 2.50.4 released!](#)
16 DECEMBER 2025
([more](#))

Security Advisories

- ▶ [WebKitGTK and WPE WebKit Security Advisory WSA-2025-0010](#)
17 DECEMBER 2025
- ▶ [WebKitGTK and WPE WebKit Security Advisory WSA-2025-0009](#)
04 DECEMBER 2025
([more](#))

Releases

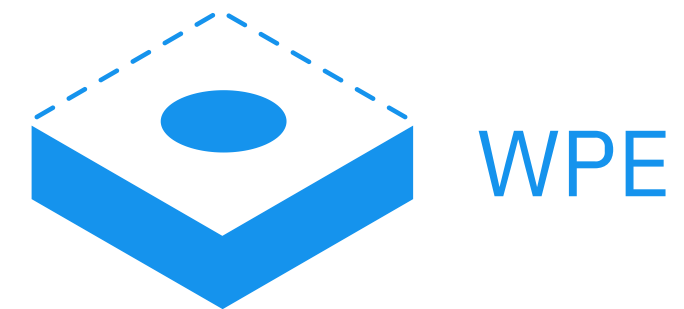
- ▶ **Stable tarball**
[WebKitGTK 2.50.4](#)
- ▶ **Unstable tarball**



What is WPE WebKit?

WPE WebKit: port of **WebKit** for Linux-based **embedded devices**

- **Shared architecture with WebKitGTK**
 - **Common parts:** GLib, Skia, GStreamer, libsoup, fontconfig...
 - **Key differences:** **no UI toolkit**, graphics stack, input handling.
- Focus on **flexibility**, **security** and **performance**
- **Minimal** set of **dependencies**
- **Backend-based** architecture for I/O
- **Low memory** and **storage footprint**
- Platform-specific **HW-acceleration**



 <https://wpewebkit.org>



What is WPE WebKit?

```
mario:~ $ wkdev-enter --name wkdev
wkdev-enter: Launch a command or spawn an interactive shell in a container built by 'wkdev-create'
```

\X /|| \ \X(_ _ \ \X \X /| (_ _ _ \X(_ _ \ \X /X
 |) (| | \ / / | (_ _) | (_ _ \X | (\) | \ \ / /
 | | _ | | (/ / | |) | | (_ _ | | | | _ _ _ | (_ _ _ | |) | | (/ /
 | | () | | _ (| | | | | _ _) (()) (_ _ _) (_ _ _) | | | | | _ (_
 | | | | | | (\ \ | |) | | (_ _ _ \ \ _ / /) | | | |) | | (\ \
 | O O | | / \ \ | (_ /) | (_ _ _ \X \ / / \X _ _ _) | | (_ /) | / \ \
 (_ _ _ _) | _ / \X (_ _ _ _ / (_ _ _ _ / \ / \X _ _ _ _) (_ _ _ _ / | _ / \X

```
Git repository 'wkdev-sdk' ( main @ 610e906829c272e3f8e035358e9b06f3f61534fc )  
→ Last commit: Thu Dec 11 23:53:57 2025 +0000
```

NOTE:

- Be sure to try `'wkdev-test-host-integration'` to verify your container setup b
- The home directory `${HOME}=/home/mario` within the container is not the same a
You can find your regular host home directory in the container under `${HOST_H`

- Instructions on how to build / debug / profile WebKit can be found in the SDK locally in `#{WKDEV_SDK}/docs` or online <https://github.com/Igalia/wkdev-sdk/tree/master/docs>

```
mario@wkdev:~$ cd /host/home/mario/work/WebKit/
```

```
mario@wkdev:/host/home/mario/work/WebKit$ ls WebKitBuild/
```

WPE

```
mario@wkdev:/host/home/mario/work/WebKit$ ls WebKitBuild/WPE/
```

Release

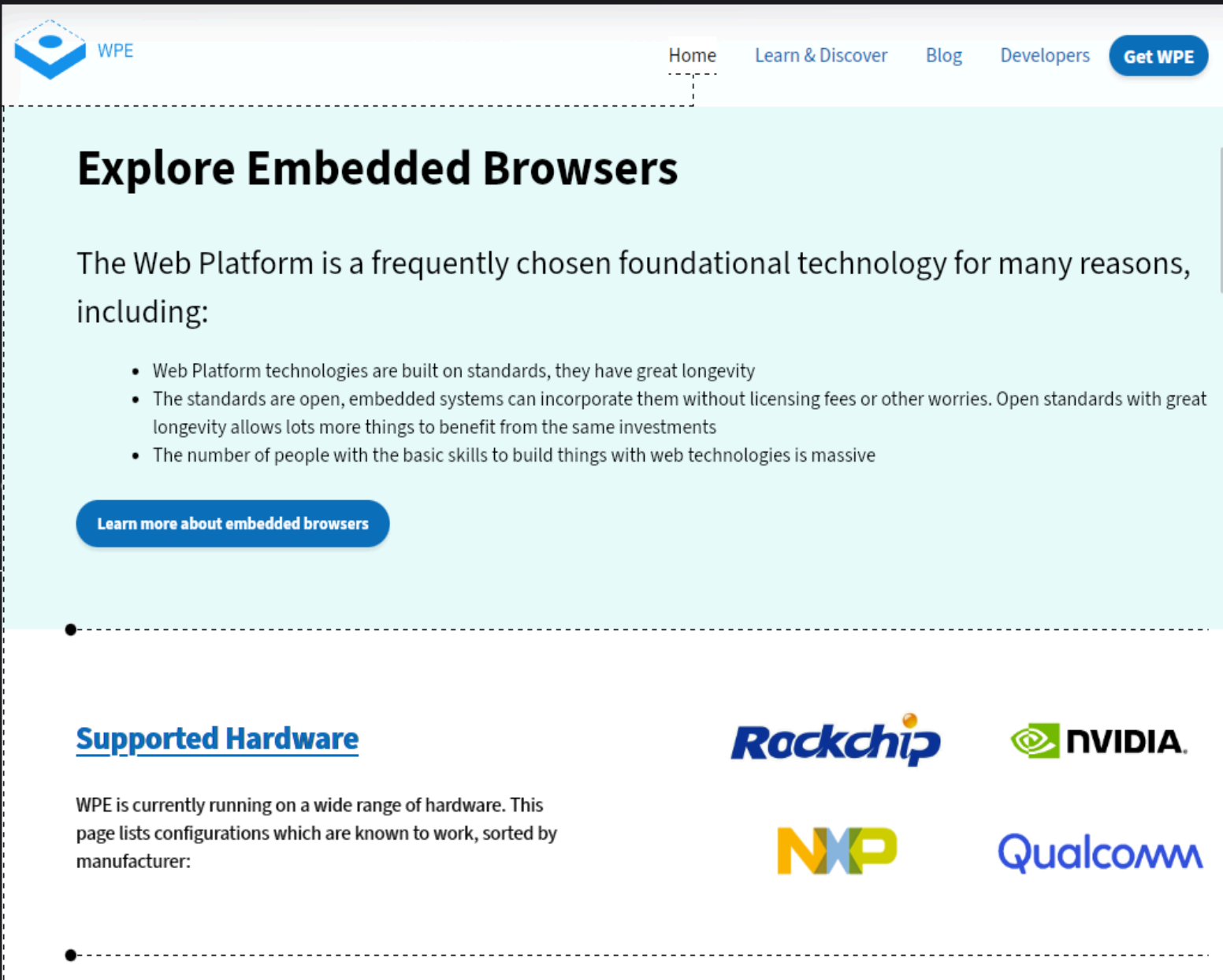
```
mario@wkdev:/host/home/mario/work/WebKit$ ls WebKitBuild/WPE/Release/
```

ANGLE	CTestTestfile.cmake	manifest.txt
bin	DerivedSources	nimlang.ops
bmalloc	DeveloperTools	PAL
build.ninja	Documentation	PerformanceTests
build-webkit-options.txt	facebook.ops	reddit_memory_warning.ops
CMakeCache.txt	flickr_memory_warning.ops	reddit.ops
cmakeconfig.h	flickr.ops	share
CMakeFiles	GI DocgenGenerated	Skia
cmake_install.cmake	inspector-resources.stamp	Source
compile_commands.json	JavaScriptCore	TestRunnerShared
CPackConfig.cmake	JavaScriptCoreGLib	theverge_memory_warning.ops
CPackSourceConfig.cmake	lib	theverge.ops

```
mario@wkdev:/host/home/mario/work/WebKit$ ./Tools/Scripts/run-minibrowser --wpe https://wpewebkit.org
```

Using default MiniBrowser

```
MESA-INTEL: warning: ../src/intel/vulkan/anv_formats.c:981: FINISHME: support more multi-planar formats with DRM modifiers
```



Why do Web rendering engines matter in embedded devices?

- **Strategic role in the software stack** of embedded devices
 - Rendering, networking, security sandbox, media, I/O, accessibility...
- The **Web Platform allows building all sorts of applications**
 - Flexibility for designing, implementing and testing your product.
- **Known development stack**
 - Massive pool of web developers that could implement applications.
- **Useful to implement all kinds of products**
 - Smart home, In-Vehicle/Flight Infotainment, digital signage...



WebKit on embedded devices

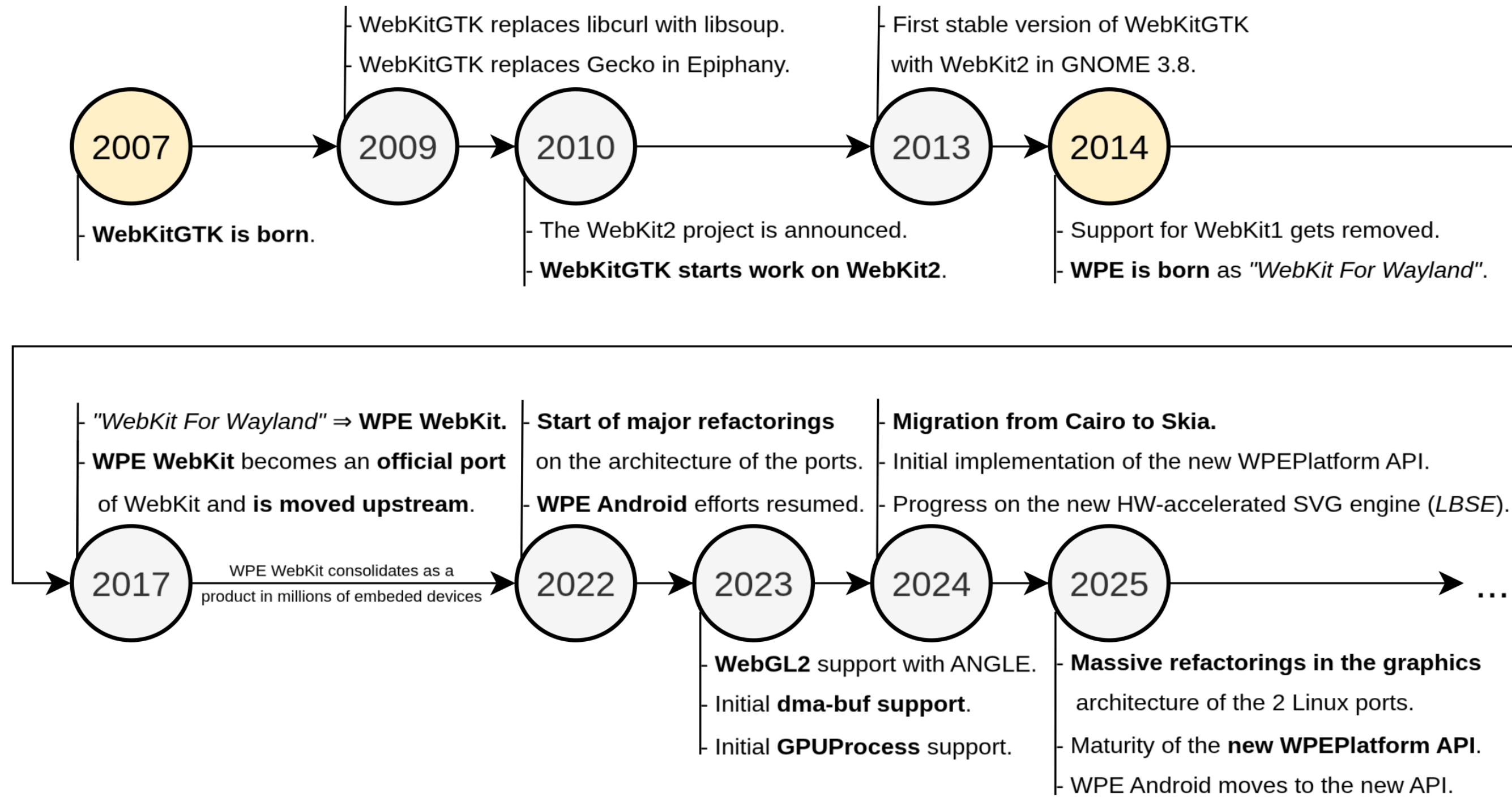
- Phones & tablets
- Set-top boxes & TVs
- Smart home appliances
- GPS navigation devices
- Audio/video conferencing
- Hi-Fi sound systems
- Audio streaming
- Digital signage
- Server-side rendering
- QA and testing



History of WebKitGTK & WPE WebKit



History of WebKitGTK & WPE WebKit



The Web Platform on Linux devices with WebKit: where are we now?

Mario Sánchez-Prada, FOSDEM 2026



Latest updates



Latest updates

- **2 new stable releases** of WebKitGTK and WPE WebKit:
 - Published in March (2.48) and September (2.50), as usual.
 - Numerous updates and improvements in HTML, CSS, JS and Web features.
- **Multimedia-related improvements**
 - WebM support in MediaRecorder, WebAudio fixes, WebCodecs compliance.
 - GStreamer-based WebRTC backends.
- **Big overhaul to the graphics rendering pipeline**
 - Major refactoring and removal of unused abstraction layers.
 - Threaded GPU rendering, GPUProcess for WebGL.
 - Damage tracking and propagation to the compositing stage.



Latest updates

MotionMark 1.3 improvements on the 32-bit & 64-bit Raspberry Pi 4

Last processed revision — RPi4 32-bit: [305815@main-RPi4](#) 64-bit: [305924@main](#)



The Web Platform on Linux devices with WebKit: where are we now?

Mario Sánchez-Prada, FOSDEM 2026



Latest updates

MotionMark 1.3 improvements on the 64-bit Raspberry Pi 4

Test	Score July 2024	Score April 2025	Score October 2025	Score January 2026
Multiply	501.17	710.75	697.15	678.93
Canvas arcs	140.24	820.64	859.68	859.48
Canvas lines	1613.93	3025.16	4648.54	7508.43
Paths	375.52	4268.87	3953.83	4288.59
Leaves	319.31	480.19	684.72	673.94
Images	162.69	265.14	263.19	267.88
Suits	232.91	444.55	388.62	399.03
Design	33.79	63.99	114.09	100.24
OVERALL	254.15	634.49	737.56	778.99



Latest updates

- **JavaScriptCore**

- Improve memory management for long-running applications.
- New tooling to detect memory leaks and do profiling.
- WASM-related work for 32-bit support (i.e. BBQJIT, OMG, IPInt).

- **Security**

- 2 major feature releases + several bug/security fixes releases.
- Increase use of smart pointers, dropped libsoup2 support.

- **Quality assurance**

- Better infrastructure, improved test coverage.
- Move the QA bots to the new **WebKit Container SDK**.



Latest updates

- **New WPEPlatform API**

- Support for accessibility, multiple views, touch events in DRM platform.
- GTK4 platform implementation via an external backend.
- Multiple API additions and API tests. Integration with QA bots.

- **Android support**

- Upstreamed all the patches from the WPE-Android project.
- AHardwareBuffer support for Android in WebKit upstream.
- Native integration with Android subsystems (e.g. logging).
- Kept evolving the WPE-Android project (WPE 2.50, NDK r27, WPEPlatform API).

- **WebXR support**

- Implemented WebXR using OpenXR and added support for the WebXR AR module.
- Enabled WebXR support on both Linux and Android.



Next steps



Next steps

- **Multimedia-related improvements**
 - GStreamer-based WebRTC backend.
 - Screen capture streaming to WebRTC PeerConnection.
 - WebCodecs integration with WebGL and WebAudio.
 - Player suspension (useful in pages with many media elements).
- **More work on the **graphics rendering pipeline****
 - Align the graphics architecture with other ports.
 - Compositor refactor to avoid OpenGL dependency and support new APIs.
 - Improvements around async scrolling animations.
 - Enable GPUProcess in more cases (i.e. WebGL only for now).
 - Remove Cairo support.



Next steps

- **JavaScriptCore**
 - Memory improvements for long-running applications.
 - Better tooling to investigate memory-related problems.
- **Security**
 - Keep working on the same release cadence and handling Security Advisories.
 - Smart pointer coverage, reduce instances of unsafe buffers access.
- **Quality assurance**
 - Better QA infrastructure and improve test coverage.
 - Resume the WebKit *gardening* efforts for the Linux ports.



Next steps

- **New WPEPlatform API**
 - Complete the new WPEPlatform API.
 - Write more API tests and documentation.
 - Release version 1.0 (aiming for WPE 2.54).
- **Android support**
 - Complete migration to the new WPEPlatform API.
 - Integrate with the upstream testing infrastructure.
- **WebXR support**
 - Implement more WebXR modules.
 - Implement WebXR layers, hit test, and anchors.



Wrapping up

- **WebKit** keeps being a **complete and embeddable Web engine on Linux**.
 - **WebKitGTK** for GTK-based applications, **WPE WebKit** for embedded devices.
- Both ports provide **mature and actively-maintained Linux ports**.
 - Big push in the past 4 years greatly improved performance and quality.
 - Recent releases delivered major improvements, particularly around graphics.
- **Next steps** focused on **performance, stability, and maintainability**.
 - **New possibilities** opening with **Android** and **WebXR** support.

It's a pretty exciting moment for **WebKit on Linux**!



How to contribute

WebKit:

Website: <https://webkit.org>

Mailing list: <https://lists.webkit.org/mailman3/lists/webkit-dev.lists.webkit.org>

Documentation: <https://docs.webkit.org>

WebKitGTK:

Website: <https://wpewebkit.org>

Mastodon: <https://floss.social/@WebKitGTK>

Bluesky: <https://bsky.app/profile/webkitgtk.org>

Mailing list: <https://lists.webkit.org/mailman3/lists/webkit-gtk.lists.webkit.org>

Matrix: [#webkitgtk:matrix.org](https://matrix.org/#webkitgtk)

WPE WebKit:

Website: <https://webkitgtk.org>

Mastodon: <https://floss.social/@WPEWebKit>

Bluesky: <https://bsky.app/profile/wpewebkit.org>

Mailing list: <https://lists.webkit.org/mailman3/lists/webkit-wpe.lists.webkit.org>

Matrix: [#wpe:matrix.org](https://matrix.org/#wpe)



Questions?

Mario Sánchez-Prada

mario@igalia.com



