New browser engines

sideshowbarker (mike)

- Before joining W3C in 2007, was on the mobile-browser product-development team at Opera Software
- For the last 20 years, working daily with implementors from all major browser projects: Safari, Firefox, Chrome, and IE/Edge
- Contributed code/patches to WebKit, Ladybird, Firefox
- Living in Tokyo

Why create a new browser engine?

Answer: Why not?

The more browser engines, the better

Let one hundred browser engines bloom!

There are many good reasons to have new browser engines as alternatives for users to Safari, Firefox, Chrome, and Edge.

For the W3C, having more implementors working on more browser engines helps us make W3C specifications better.

New browser engines

- Servo
- Ladybird
- Flow (Ekioh)
- *[Your new browser-engine project here]*

Servo

- Rust , open source
- ~35 committers/month(Chrome: 1200, WebKit: 150)
- JavaScript engine is SpiderMonkey (C++) Boa?
- Huawei
- Goals? (Mobile? Performance?)

Servo: Challenges

- Missing many, many core features (CSS…)
- https://wpt.fyi/results/?product=servo
- Passing only 80% of WPT test suite (CSS…)
- Performance (giant gap vs Chrome)

Ladybird

- C++ and Swift …, open source
- ~48 committers/month(Chrome: 1200, WebKit: 150)
- Goals? (Not focusing on mobile, …)

Ladybird: Challenges

- Missing many, many core features (CSS…)
- https://wpt.fyi/results/?product=ladybird
- Passing only 89% of WPT test suite (CSS…)
- Performance (giant gap vs Chrome)
- Money/funding…

Flow

- C++, not open source
- https://wpt.fyi/results/?product=flow
- Passing only ~90% of WPT test suite (CSS…   )
- Raspberry Pi, embedded use cases, parallel layout
- *Very* small development team

The W3C needs you

How you can help

- Contribute code patches to browser engines (time…)
- Raise bugs and PRs/patches against specs
- Become a spec editor (eventually) ex: SVG
- Write and refine WPT tests (HTML and JavaScript)
- Write/translate docs for MDN

Teach/learn how browser engines work

https://browser.engineering

As you read and work through the chapters in the Browser Engineering book, you will actually end up creating a complete functional web browser in Python (though not production-ready at all).

If you are a university professor, consider adding a browser engineering course to your curriculum.

Get involved!

Thanks!

Q & A