The Web APIs powering the Starshot trial

/u/mglaman @mglaman@phpc.social /in/mattglaman

#DrupalCon

Leveraging **native** and **open** features for a unique trial experience

WebAssembly

IndexedDB API

Web Workers API

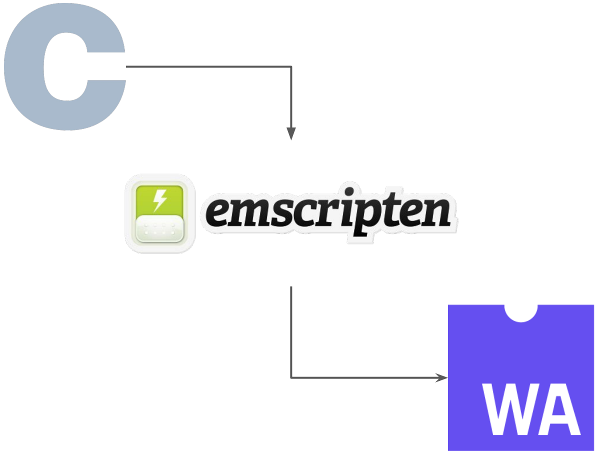
Web Components

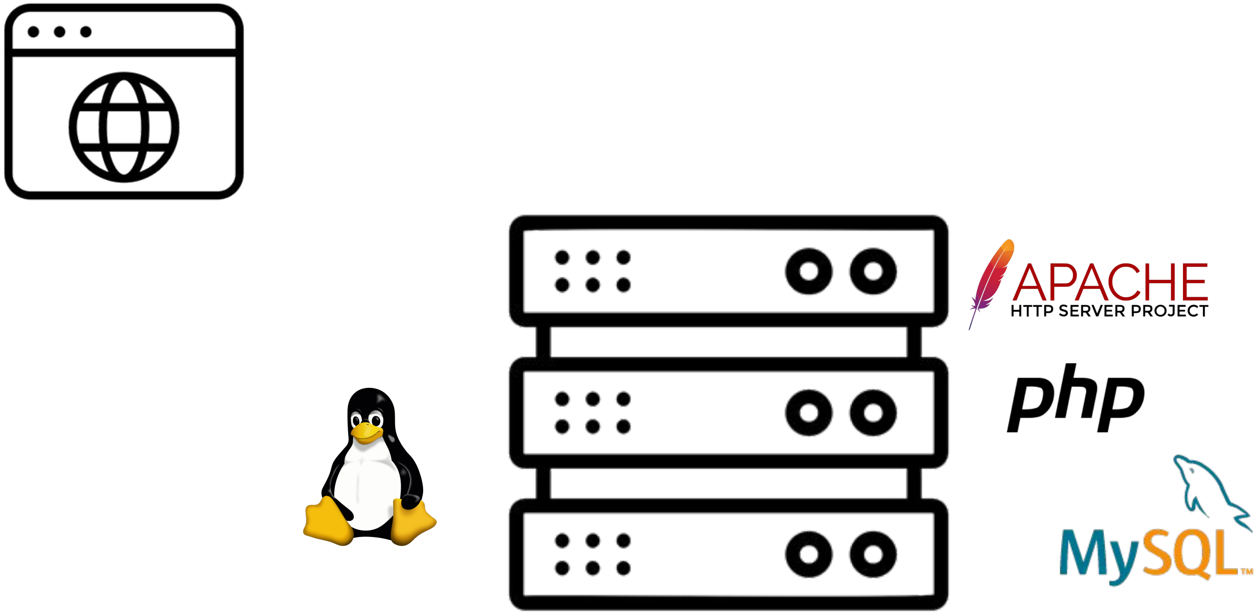
WebAssembly

PHP… in the browser?

WebAssembly (Wasm)

- Defines a portable binary-code format (like assembly language)
- A compilation target of C/C++, Rust, and Go
- Emscripten allows compiling C code, like the PHP interpreter, to Wasm
- All browser JavaScript engines support WebAssembly
- It is an open standard and managed by W3C Community Group and W3C Working Group





Drupal running in a normal stack

Drupal running in WebAssembly

WebAssembly **removes** the need for **third-party services** to try Drupal.

WebAssembly & JavaScript

- Emscripten provides JavaScript for interacting with PHP in WebAssembly using `ccall`

- Enables building “SDKs” for easily calling these functions and building applications

- Emscripten provides a filesystem API through JavaScript

- Browser is the web server by “hosting” the files

- Service worker like CGI in sending request to PHP and returning server side rendered content

```
import PhpBinary from './php-web.mjs';
return new PhpBinary()
.then(({ccall, FS}) => {
 ccall(
   'phpw_run',
   'string',
   ['string'],
   [`<?php echo "Hello, world!";`]
 );
})
```

Example invoking Emscripten **ccall** directly

```
const { PhpWeb } = await import('PhpWeb.mjs');
const php = new PhpWeb;

const exitCode = await php.run(
   '<?php echo "Hello, world!";'
);
```

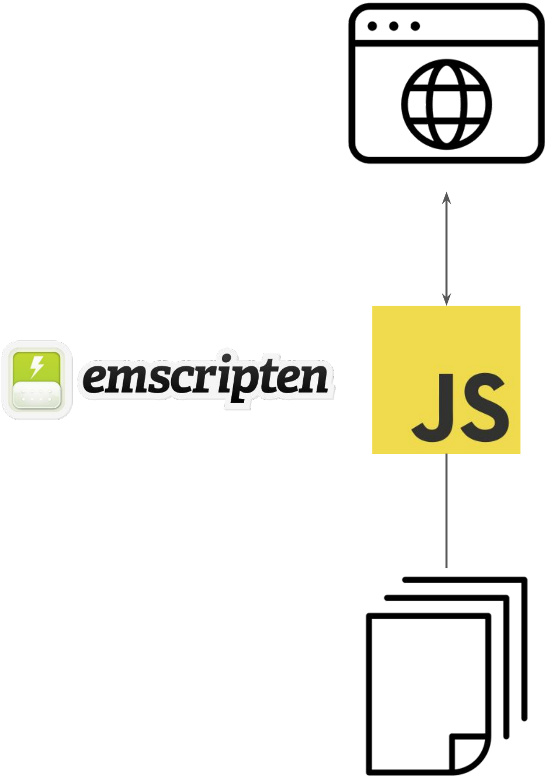
Example printing “Hello world” using **php-wasm** wrappers

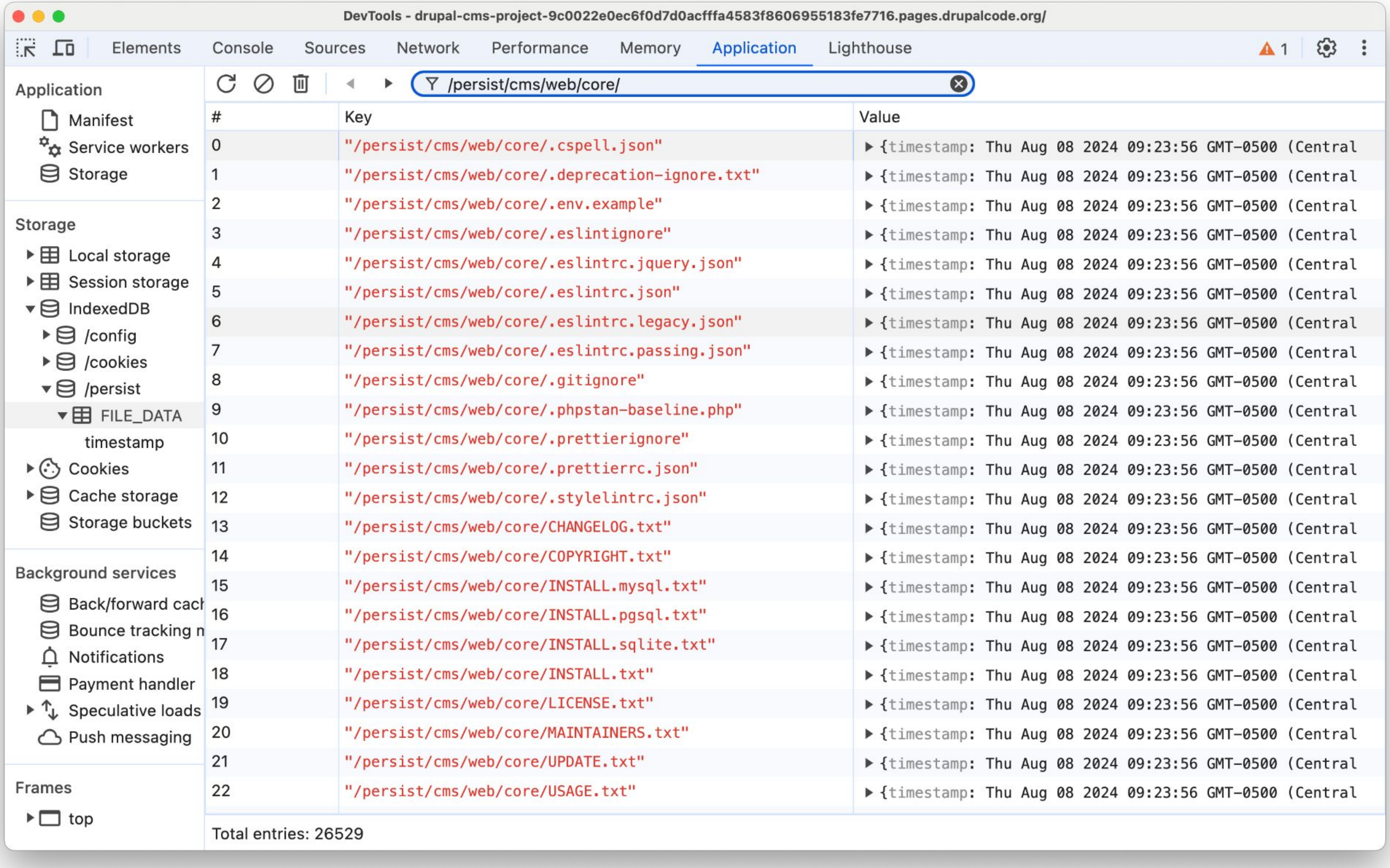
IndexedDB

Filesystem for Wasm

IndexedDB

- IndexedDB, intended for structured data, including files/blobs
- Emscripten provides a filesystem API through JavaScript to use IndexedDB
- Files are accessible by the Wasm runtime and available to PHP
- Drupal codebase is stored on the browser inside of a IndexedDB store
- IndexedDB is also being used to store cookies are as well for the user session.





DevTools - drupal-cms-project-9c0022e0ec6f0d7d0acfffa4583f8606955183fe7716.pages.drupalcode.org/

/persist/cms/web/core/

#	Key	Value
0	"/persist/cms/web/core/.cspell.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
1	"/persist/cms/web/core/.deprecation-ignore.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
2	"/persist/cms/web/core/.env.example"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
3	"/persist/cms/web/core/.eslintignore"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
4	"/persist/cms/web/core/.eslintrc.jquery.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
5	"/persist/cms/web/core/.eslintrc.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
6	"/persist/cms/web/core/.eslintrc.legacy.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
7	"/persist/cms/web/core/.eslintrc.passing.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
8	"/persist/cms/web/core/.gitignore"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
9	"/persist/cms/web/core/.phpstan-baseline.php"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
10	"/persist/cms/web/core/.prettierignore"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
11	"/persist/cms/web/core/.prettierrc.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
12	"/persist/cms/web/core/.stylelintrc.json"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
13	"/persist/cms/web/core/CHANGELOG.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
14	"/persist/cms/web/core/COPYRIGHT.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
15	"/persist/cms/web/core/INSTALL.mysql.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
16	"/persist/cms/web/core/INSTALL.pgsql.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
17	"/persist/cms/web/core/INSTALL.sqlite.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
18	"/persist/cms/web/core/INSTALL.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
19	"/persist/cms/web/core/LICENSE.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
20	"/persist/cms/web/core/MAINTAINERS.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
21	"/persist/cms/web/core/UPDATE.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central
22	"/persist/cms/web/core/USAGE.txt"	▶ {timestamp: Thu Aug 08 2024 09:23:56 GMT-0500 (Central

Total entries: 26529

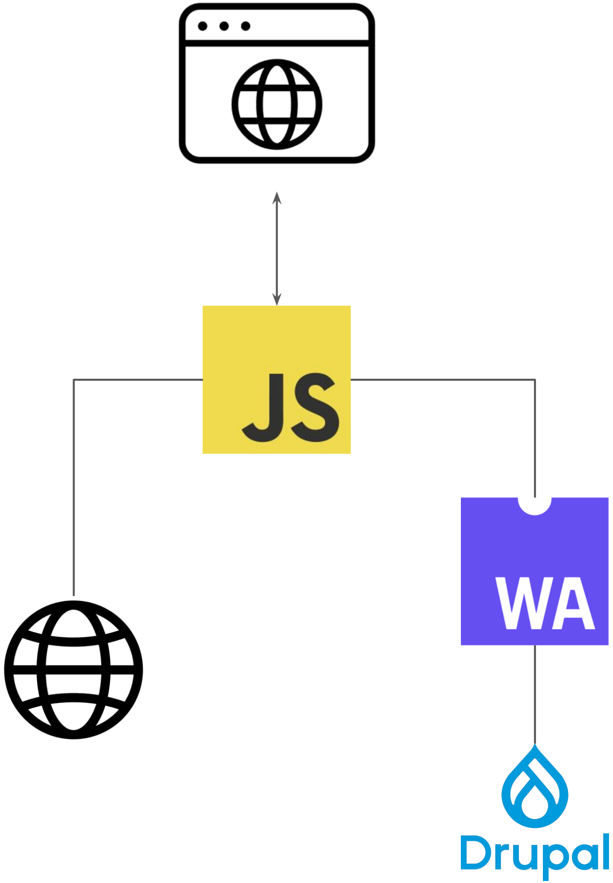
Web Workers API

Dedicated workers

Service workers

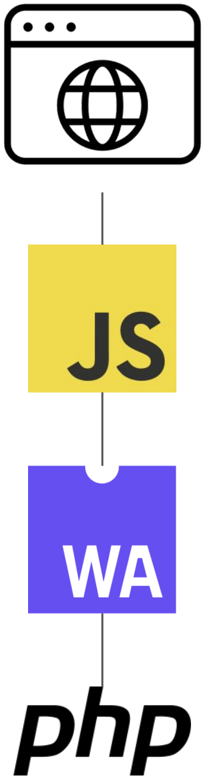
Service worker

- Service workers run in the background and can intercept outgoing fetch requests
- Instead of making a “real” request, the request can be sent to PHP in WebAssembly
- Processes the response from PHP and returns the to the browser
- Acts like CGI to return data from filesystem or server side rendered content in PHP
- PHP CGI build doesn’t allow running arbitrary code but only scripts on the filesystem



Dedicated worker

- Allows executing PHP code directly without blocking the main thread
- Prevents the browser from “locking up” and not repainting UI changes
- Used for unarchiving Drupal CMS or performing behind the scenes actions
- Unfortunately each worker is isolated, so the shared libraries for PHP get requested twice (but response disk caching should avoid double downloads)



Web Components

Custom elements

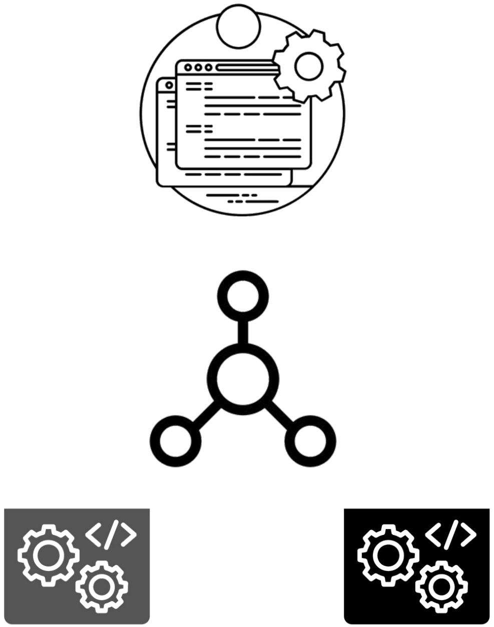
Web Components

- trial-manager web component is used to automatically start a trial or perform actions on an
- Allows for dynamic interface without using React, Vue, etc
- Web Components classes allow encapsulating logic and reacting to attribute changes
- Interacts with the dedicated worker to download and extract Drupal CMS

Broadcast channels

Broadcast channel

- Only the main window can (easily) send messages to the service worker
- Used to provide asynchronous communication between the web component, dedicated worker, and service worker
- When the service worker is ready, dispatches a message so the trial manager web component can start
- Web Component tells the service worker to refresh and recognize new files after the setup



Try it out!