Bundle Preloading

IETF WEBPACK

Felipe Erias

Igalia, in partnership with EyeO

November 2021

Introduction

- Felipe Erias
  - **felipeerias@igalia.com**
  - `felipeerias` in GitHub, Twitter

igalia

Bundle preloading

- **WICG/bundle-preloading**
- Website declares resources that it wants to preload
- A request is sent to the server
  - new header: list of resources not in cache
- Server sends a Web Bundle with the resources
- Each resource can be cached separately
- Origin and path restriction
- URL consistency: same data individual/bundled

Bundle preloading: spec

- Two related approaches:
  - Bundle preloading **WICG/bundle-preloading**
  - Packaging Websites **WICG/webpackage**
  - **comparison**
- Open issue to converge the preloading specs: **https://github.com/WICG/webpackage/issues/699**

igalia

Updated Bundle spec

- WPACK — IETF 111 (July 2021)
  - removed `primary` URL section
  - removed `manifest`
  - removed content negotiation
- Remaining sections: `index`, `critical`, `responses`

igalia

Tool support (1/2)

- Node.js module for creating and reading Web bundles
  - **npmjs.com/package/wbn**
  - source: **WICG/webpackage/js/bundle**
- Updated to the latest version of the spec (`"b2"`)
  - Backwards compatible with the previous (`"b1"`)
  - See **4.1.2. Draft version numbers**
- Suitable for prototyping a bundle-aware server

igalia

Tool support (2/2)

```
// The latest spec version is used by default
const newBuilder = new wbn.BundleBuilder();

// Or we can specify a spec version
const oldBuilder = new wbn.BundleBuilder('b1');

// Decoding a bundle will pick the right version
var myBundle = new wbn.Bundle(buffer);
console.log('Version ' + bundle.version);
```


igalia

In the browser

- The `WBN` module can also be used in the browser
  - see **client/package.json** for details
- We can prototype bundle preloading!
  - **WICG/bundle-preloading/prototypes**

igalia

Prototype

- Client parses `<script type=bundlepreload>`
- Sends HTTP request with `Bundle-Preload` header
  - list of resources not in cache
- Bundles are served by a small Web server that
  - supports bundle preloading with subsetting
  - is able to create bundles on the fly

igalia

Prototype: syntax

```
<script type="bundlepreload">
    {
        "source": "assets/img.wbn",
        "resources": [
            "img/01.png",
            "img/02.png",
            "styles.css",
            "sidebar.js"
        ]
    }
</script>
```

Prototype: examples

- Basic test
  - **prototypes/client**
  - two `<script type=bundlepreload>` tags, using absolute and relative URLs
- *Create React App*
  - **prototypes/examples/create-react-app**
  - integrated with React/webpack
  - fewer requests, parallelization

Prototype: insights

- Try out realistic scenarios (with some setup work)
- Try out different syntaxes
  - `<script>` tag and HTTP headers
- Performance is reasonable
  - reduced number of requests/responses
  - resources can be fetched in parallel
  - resources are cached individually
  - (but service worker adds overhead)

Prototype: next steps

- Better integration with bundler tools (e.g. webpack)
- Early benchmarking results
  - comparison with bundler tools and other solutions
- Create more examples
  - ideas welcome!
- Can this approach be applied for other use cases?

Thank you!

Felipe Erias

felipeerias@igalia.com

igalia