Web Bundles as content packages

A web bundle, formally known as a [Bundled HTTP Exchanges](#) is a set of resources packaged together for distribution with a `.wbn` extension and `application/webbundle` mime type.

The bundle contains multiple resources that make up your website. These range from HTML, CSS and Javascript to svg, audio and other elements necessary to make your site or applicatioon work.

Bundles don't depend on users accessing the site or app once before it will work offlline. The bundle contains everything necessary for the site or application to run offline.

Bundles retain the interactivity of the original site because it allows Javascript to run.

Having a single package for an application opens new avenues for distribution. You could share your content from USB sticks or other promotional media... The team at the Chrome Developer Summit had a USB stick hidden inside a Chrome Offlline Dinsoaur shape.

Getting Started

Currently the easiest way to create unsigned web bundles is to use the [go/bundle](#) CLI. To install the CLI (and Go if necessary) follow the instructions in [Building Web Bundles](#)

Once you have the go/bundle installed build your content and run the following command; make sure you replace the `-baseURL` and `primaryURL` values with valid URLs that work for your site or application.

```
gen-bundle \
-dir docs \
-baseURL https://layout-experiments.firebaseapp.com/ \
-primaryURL https://layout-experiments.firebaseapp.com/ \
```

```
-headerOverride 'Access-Control-Allow-Origin: *' \
-o layout.wbn
```

The `layout.wbn` is the bundle ready to open.

Chrome version and flags

Web bundles will only work on Chrome 80, in Canary at the time this post is written. **The feature is on an [Origin Trial](#) so you shouldn't rely on it for production.**

Open `chrome://flags/#web-bundles` and set the flag to enabled.

Now you can open the Bundle in Chrome.

Taking size into account...

One thing to consider is the size of the bundles. A bundle for [Layout experiments](#) is 21MB while a bundle for Google's [web.dev](#) is around 212MB.

One solution to this is to use a combination of the [Network Information API](#) and offline events to only display download data when in WIFI connectivity.

In an ideal world, the code might look like this:

```javascript
if ('connection' in navigator) {
  if ((navigator.connection.type === 'wifi') ||
      'wifi'ator.connection.type === 'ethernet')) {
    'ethernet'// display dowonload bundle section
    console.log('Using wifi or Ethernet');
  } else {
    console.log('Not on Wifi or Ethernet');
  }
}
```

Unfortunately the reality is far from ideal. As currently implemented the API doesn't reflect the actual connection but reports unknown for all connections. This

defeats the purpose of using it.

Resources

- Get started with Web Bundles
- Using Web Bundles
- go/bundle