

Web Bundles

IETF 107 WPACK

Jeffrey Yasskin

[draft-yasskin-wpack-bundled-exchanges-02](#)

General Semantics

A bundle is a sequence of bytes, possibly retrievable from a URL.

Contains a set ("cache"?) of HTTP-like [resources](#):

- Each resource is addressed by a URL.
- Can do some content negotiation to select a [representation](#).

Has some bundle-wide metadata:

- Identifies the "primary" resource.
- + W3C App Manifest, EPUB package document, etc.

Establishing Authority

A resource inside a bundle, named `https://someauthority/path`, might be **authoritative** for that URL, in the sense defined by HTTP.

- If the bundle was retrieved from `https://someauthority/bundle`.
- Other mechanisms discussed in the subsequent presentations.

HOW STANDARDS PROLIFERATE:
(SEE: A/C CHARGERS, CHARACTER ENCODINGS, INSTANT MESSAGING, ETC.)



Format properties

- Secure
- Extensible
- Random-access
- Streamable loads

Format property: Secure

Minimize probability of implementation errors.

For example: ZIP allows implementations to read a resource name from the wrong place.

Easy to parse, few options, & no redundant information

Format property: Extensible via "sections"

Extensible by defining new metadata section types, which appear before resource bodies.

Signatures are currently done as an extension section.

Format property: Random access

Minimize the number of bytes to read before the client can use an arbitrary resource.

Format property: Streamable loads

Minimize the size of the prefix needed before the client can start using the first contained resource.

Starting with a full index compromises this.

Format non-property: Streamable generation

If a derived value (like a length) is computed from some content, the value must be serialized *after* the content.

Requires at least one of:

- Non-streamable loading
- Non-random access
- Redundant information
- Options in the format

Origins and URLs

If `https://my.example/bundle` contains a resource named `https://claimed.example/content`, what URL names that resource?

`package:https:,,my.example,bundle$https:,,claimed.example/content` 🤔

What is that resource's origin?

`package:https:,,my.example,bundle$https:,,claimed.example` 🤔
`ni:///mi-sha256;mi-sha256(content)` 🤔

Implementation status in Chromium

Navigate to a bundle: try it behind `chrome://flags/#web-bundles`:
<https://web.dev/web-bundles/>

Loading subresources from a bundle: being prototyped and measured.

Questions

- Within a bundle, do we need to address by arbitrary URLs, or just paths?
- Do we need content negotiation within a bundle?
- Separate streamable format from random-access format?
- URLs and origins for resources in bundles???