Streamed Bundle Generation

Jeffrey Yasskin
WPACK — IETF 108
2020-07-31
WICG/webpackage#577

Status Quo in draft-yasskin-wpack-bundled-exchanges

- Index
  - Resource 1 name, offset, size
  - Resource 2 name, offset, size
  - Resource 3 name, offset, size
  - ⋮
- Resources
  - Resource 3
  - Resource 1
  - Resource 2

Supports random and streamed loading

For simplicity, neither is quite as fast as optimal:

Random access

Read on average half of the index before jumping to resource.

B-trees could improve this.

Streamed loading

Read names of all resources before bodies start.

Could interleave names with bodies.

No support for streamed generation

Generator has to know names and sizes for all resources before sending the index.

Fix by allowing generator to send more names after the first batch?

Use cases

- Preloading subresources that originate from a variety of different-speed backends.
  - Ads, where some auctions complete faster than others.
- Optimize streamed loading time when we don't need random access.

Design sketch

- Add a second variant of the format
- Without support for random access.
- Roughly match HTTP/2 PUSH_PROMISE:
  - Blocks of resource names tell a client not to fetch that name directly.
  - Following blocks provide resource bodies.
- Signature- and hash-based identities come with the names.

Worth doing?

Pro

- Serves some extra use cases.
- Can probably re-use some components
- Optimizes streamed loading.

Con

- Adds implementation complexity.
- Duplicates MHTML.
- Random-access users need to transcode.