Weby budoucnosti s AMP

Jak AMP z roku 2015 ovlivnil rok 2020

TopMonks Caffè 2020-12-01

@robinpokorny

AMP —> 2020

Superrychlé přednačtení

Výkonné obrázky

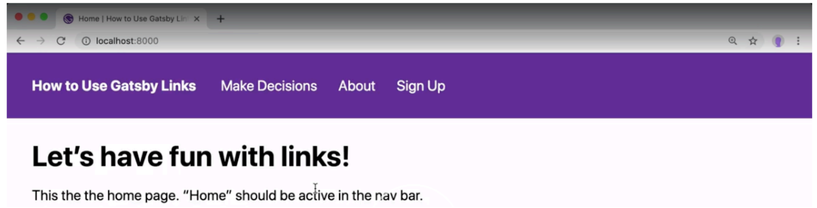
Gatsby Link API

For internal navigation, Gatsby includes a built-in `<Link>` component as well as a `navigate` function which is used for programmatic navigation.

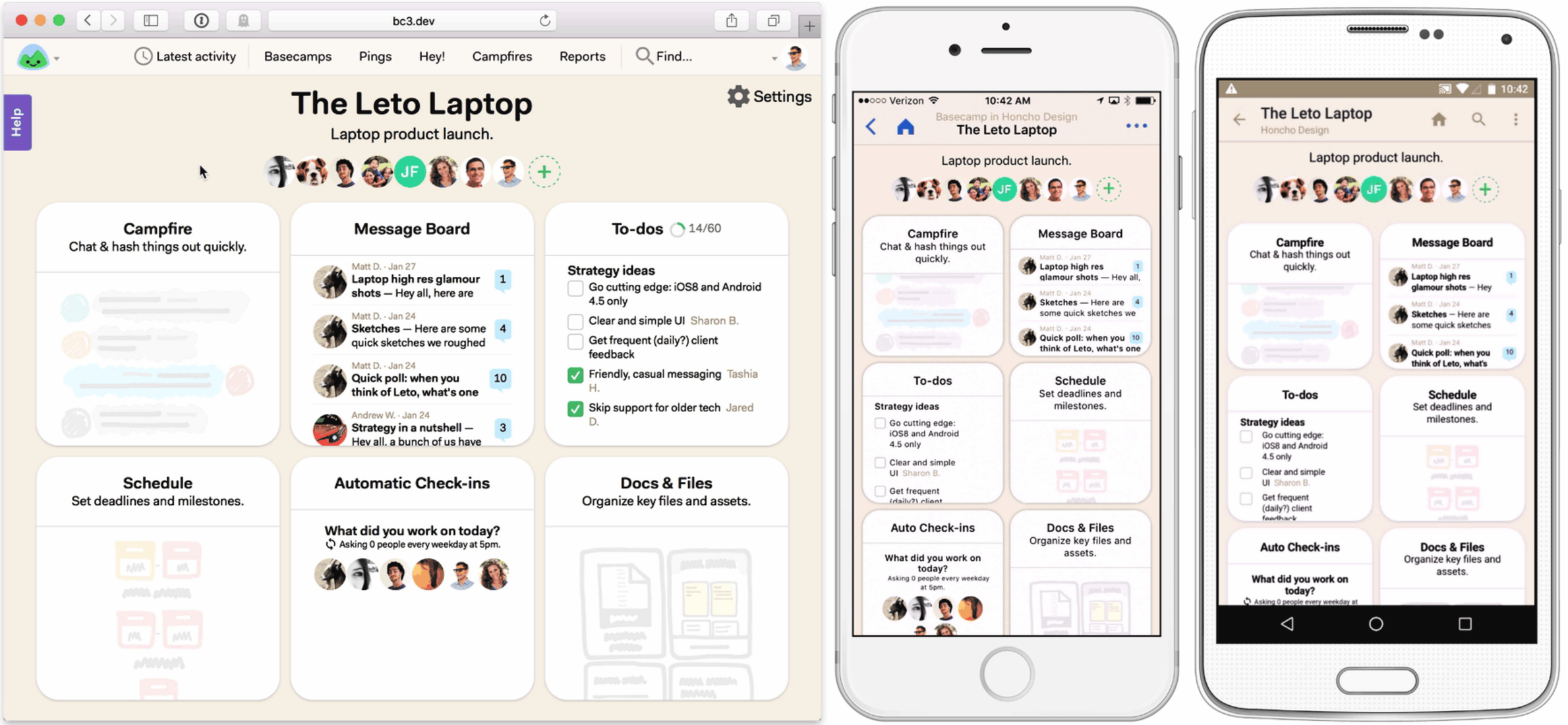
Gatsby's `<Link>` component enables linking to internal pages as well as a powerful performance feature called preloading. Preloading is used to prefetch resources so that the resources are fetched by the time the user navigates with this component. We use an `IntersectionObserver` to fetch a low-priority request when the `Link` is in the viewport and then use an `onMouseOver` event to trigger a high-priority request when it is likely that a user will navigate to the requested resource.

The component is a wrapper around @reach/router's Link component that adds useful enhancements specific to Gatsby. All props are passed through to @reach/router's `Link` component.

How to use Gatsby Link



https://www.gatsbyjs.com/docs/gatsby-link/



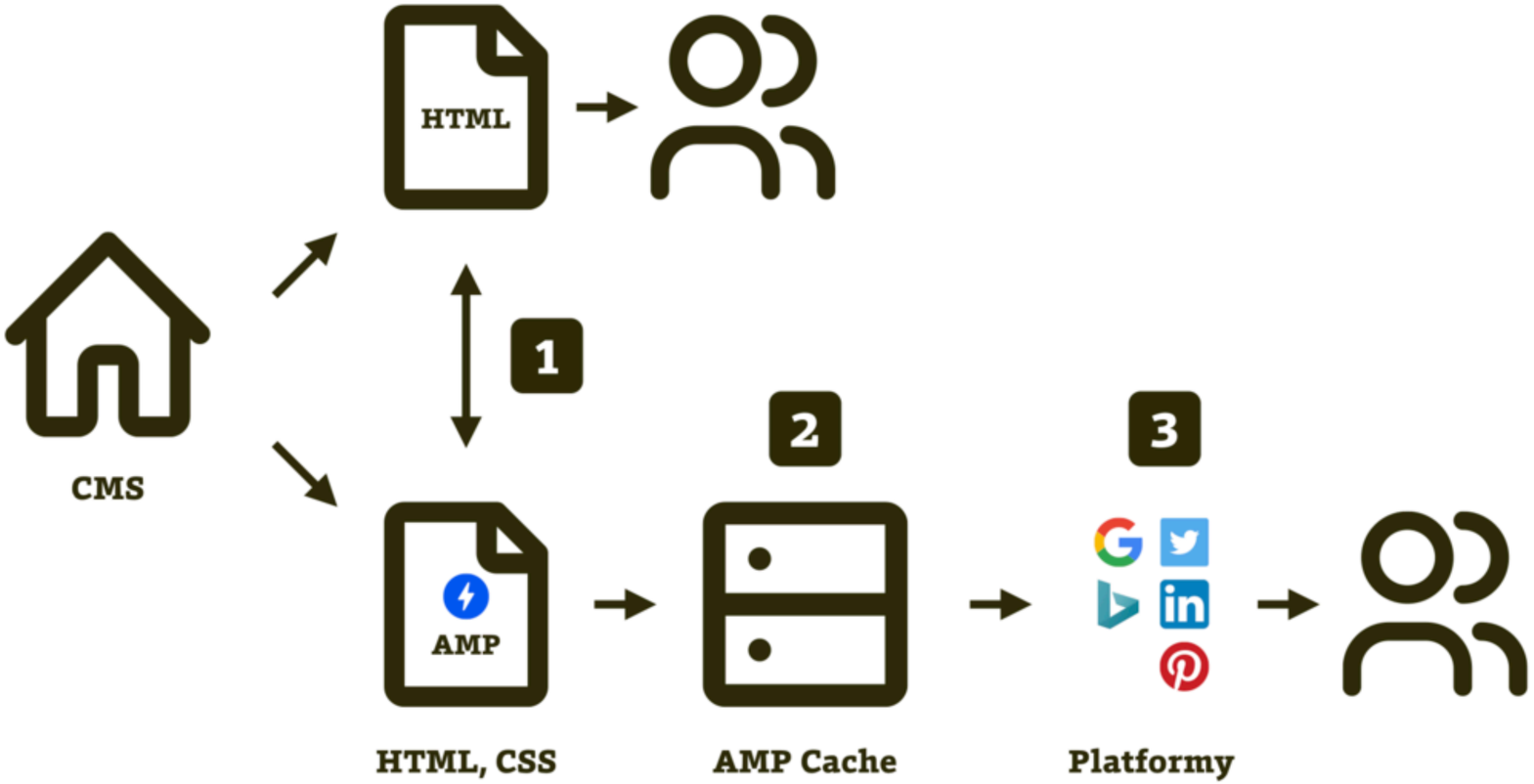
https://github.com/turbolinks/turbolinks

Cross-origin

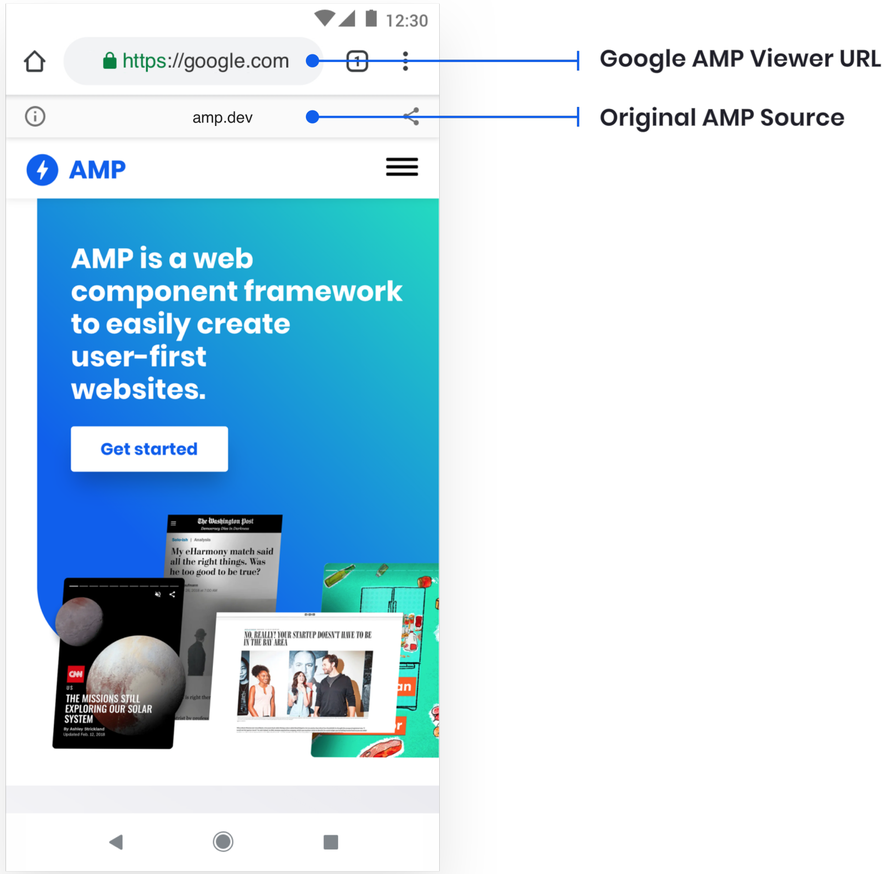
Problémy přednačtení

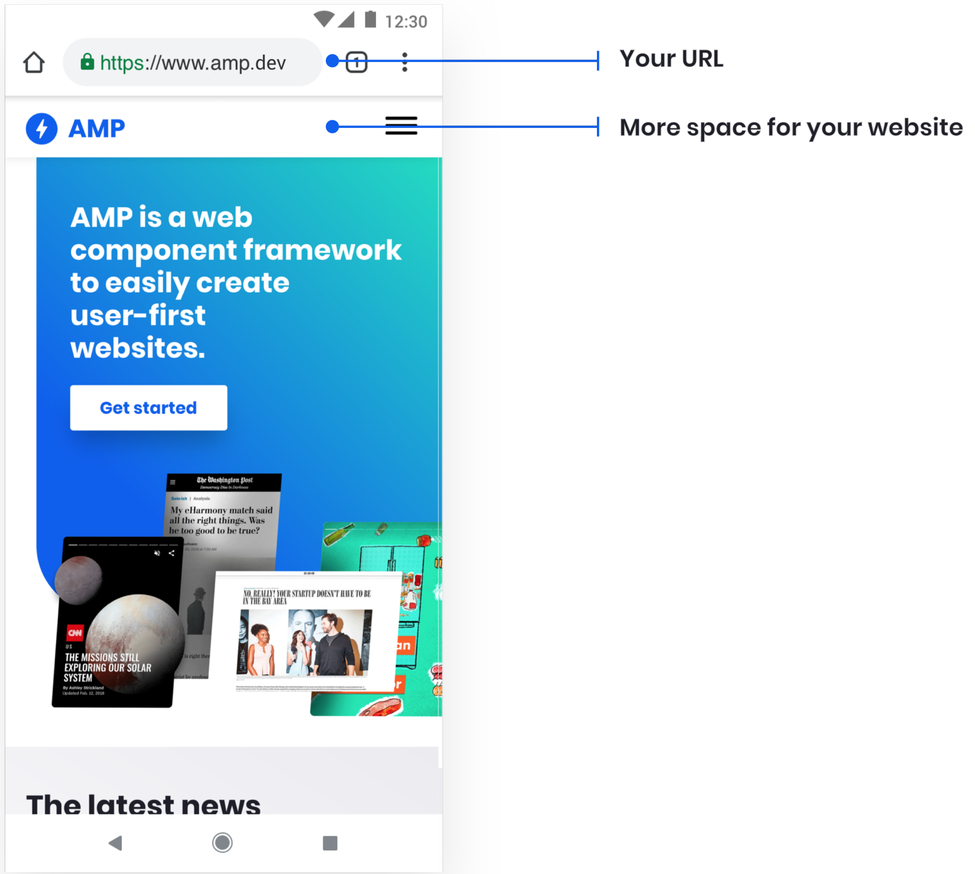
Soukromí

Bezpečnost



https://www.vzhurudolu.cz/prirucka/amp





https://developers.google.com/search/docs/guides/about-amp

Web packages

Web Bundles

Signed exchanges (SXG)

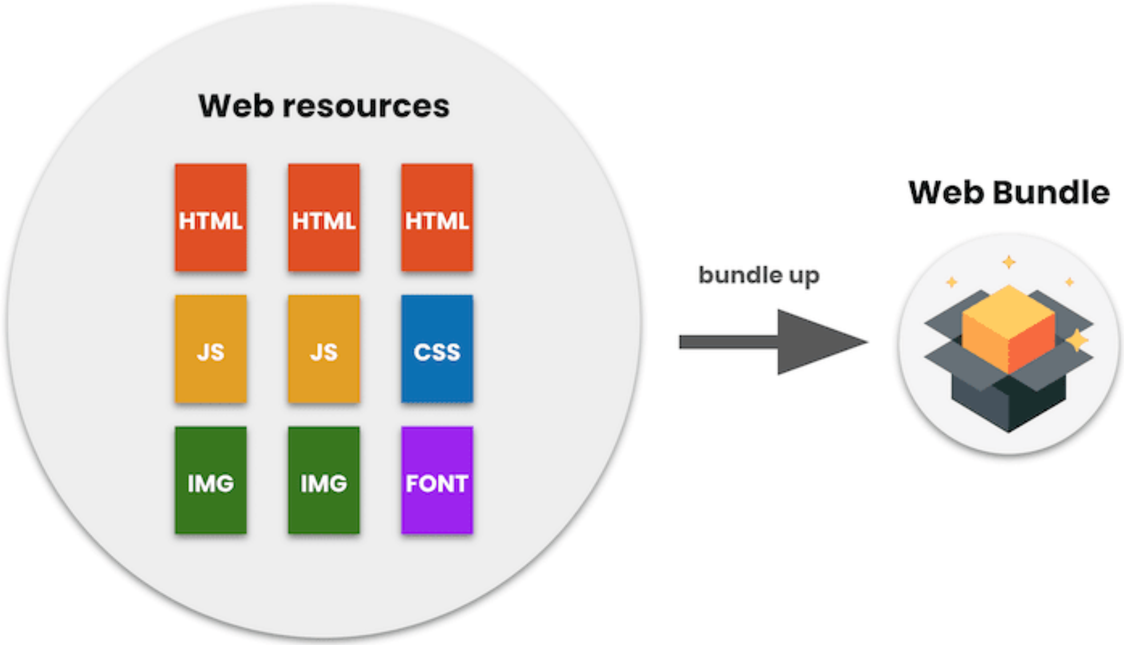
Nový fetch()

Web packages

Web Bundles

HTTP req + res v ZIPu

IETF



Signed exchanges (SXG)

Nový fetch()

https://wicg.github.io/webpackage/draft-yasskin-wpack-bundled-exchanges.html

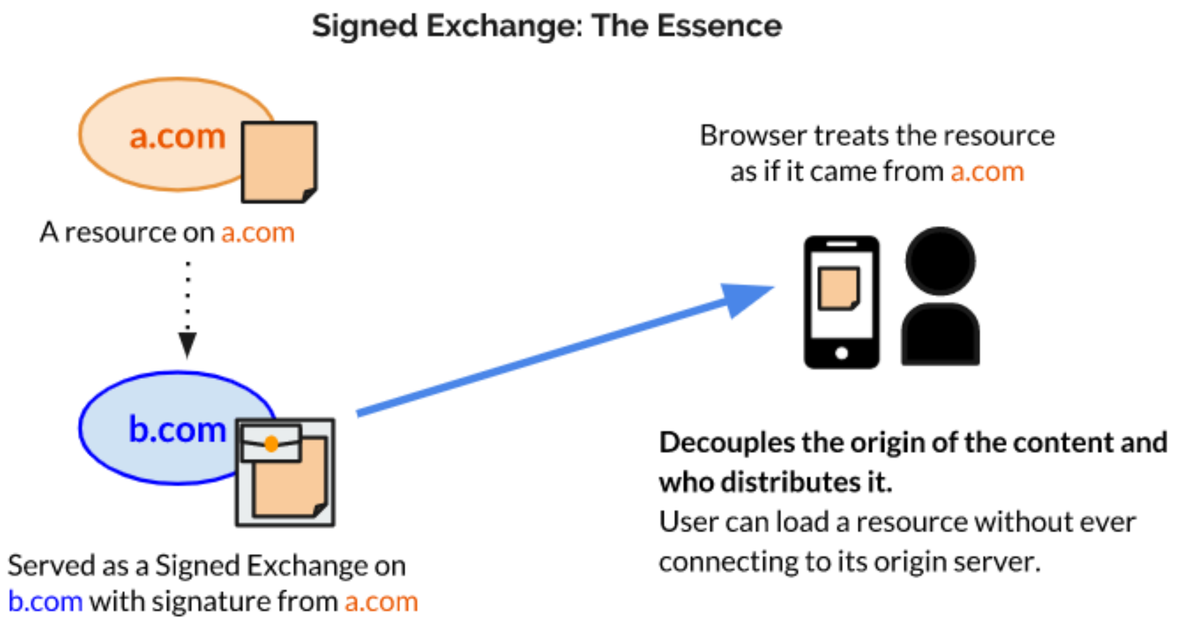
Web packages

Signed exchanges (SXG)

Vyžaduje speciální certifikát

Pro prohlížeč jako z cache

Plynule k PWA



Web Bundles

Nový fetch()

https://wicg.github.io/webpackage/draft-yasskin-http-origin-signed-responses.html

Web packages

Nový fetch()

Nejdříve balíček, pak internet

Web Bundles
Signed exchanges (SXG)

https://wicg.github.io/webpackage/loading.html

Největší změna od Web Workers

Absolutní decetralizace

Co s tím?

- Offline prohlížení
- Offline instalace
- Přednačtení se soukromím
- Obcházení cenzury
- CDN bez sdílení TLS

Mozilla's Position on Web Packaging

Web packaging proposes a significant change to the web platform in the way that content is delivered and authenticated.

From a technical standpoint, the changes are thorough and well-considered. There are some technical costs around security, operations, and complexity, but the specifications take steps to limit most of these costs.

The most disruptive feature of the proposal, origin substitution, describes a fundamental change to the security architecture of the web. In addition to a significant increase in complexity, origin substitution creates new angles of attack that site operators need to consider before they adopt the technology. Changes to the way sites operate could result in non-trivial security risks.

The main concern is web packaging might be employed to alter power dynamics between aggregators and publishers. At this moment, we don't understand enough to say definitively that this is damaging to the system. How this technology is deployed matters. Deployment without systems of accountability, oversight, and limitations on use could be harmful. There are no constraints on deployment in the proposals, so much depends on how the technology is used and the incentives around that use.

As a large suite of mechanisms, there are parts of web packaging that could be valuable on their own, such as the design of a common resource bundling format. There are also

Výkonné obrázky

Známá velikost

Lazy loading

Images on the Web

While our focus on reducing the amount of JavaScript the browser has to load has paid off, the web is not only Javascript: it's also markup and images.

Images take up 50% of the total bytes on web pages.

Images have a big impact on Largest Contentful Paint as they're often the largest visible element when a page is loaded. Largest Contentful Paint is a Core Web Vital that Google will be using in their search ranking very soon.

Half of all images are over one megabyte in size, which means they aren't optimized to be displayed on the web.

Nowadays users browse the web using their phones, tablets, and laptops, yet images are still as a one size fits all. For example: sites load a 2000 by 2000 pixel image, but phones are only displaying it as 100 by 100 pixels.

Furthermore, 30% of images on web pages are outside of the initial viewport, meaning the browser loads images that a user does not see until they scroll further down the page.

https://nextjs.org/blog/next-10


AMP
CHARON
N°57
TRANSCONTINENTAL
@robinpokorny