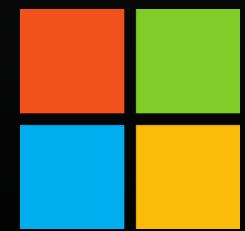


Media in the Age of PWAs



Aaron Gustafson
@AaronGustafson
slideshare.net/AaronGustafson

What exactly is a
PWA?

What exactly is a
Progressive Web App?

119

20

40



DISCUSS (51)

...



What the heck is a "Progressive Web App"? Seriously.



Ben Halpern Nov 6 '17 Updated on Mar 21, 2018 · 3 min read

#webdev

This morning I went to write an article outlining a few things about

**Ben Halpern**

@ben

A Canadian software developer who thinks he's funny.

+ FOLLOW

work

Founder/webmaster at DEV

location

Brooklyn, NY

education

Mount Allison University

joined

Dec 27, 2015



Seriously, though. What is a progressive web app?



Amberley Romo [Follow](#)

Aug 6, 2018 · 8 min read



Infrequently Noted

Alex Russell on browsers, standards, and the process of progress.

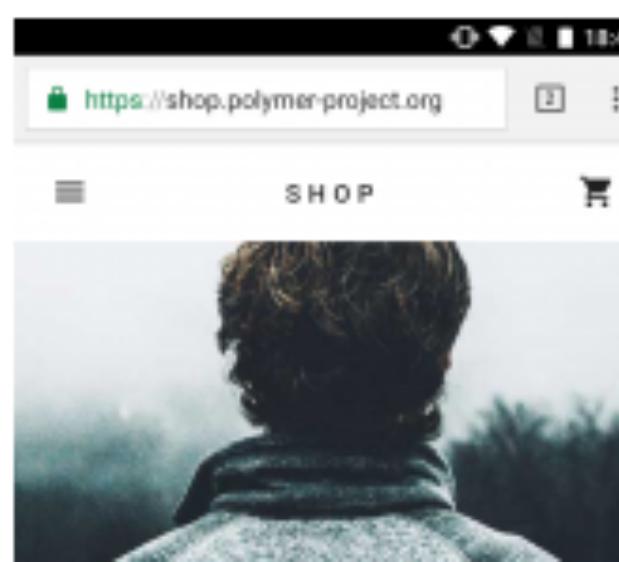
[« PWA Discovery: You Ain't Seen Nothin Yet](#)

[Web Components: The Long Game »](#)

What, Exactly, Makes Something A Progressive Web App?

Since [Frances](#) and I published a blog post last year introducing [Progressive Web Apps](#), a healthy conversation has started about what is and isn't a PWA. There are a lot of opinions and many shades of gray. What are the hard requirements? Which requirements are marginal? What's aspirational? This article outlines these requirements, attempts to classify them, and provides a baseline for "what is a Progressive Web App?"

Browsers gate Progressive Web App installation prompting and badging on criteria that they detect when users navigate to sites. These criteria have been designed to ensure that sites which invoke prompts are reliable, fast, and engaging.



What's All This Then?

I'm Alex Russell, a [web developer](#) working on Chrome, Blink, and the Web Platform at Google. I'm guilty of [many JavaScript transgressions](#).

I help lead the team building a new application model for the web, and serve on ECMA TC39 (the standards body for JavaScript). I'm an elected member of the W3C Technical Architecture Group and am Tech Lead for Standards inside the Chrome team. I design and advocate for extensible, layered, data-driven evolution of the web platform.

My professional aim is to make the web a better platform and to the extent that I can keep politics and economics from [creeping in](#).

“Progressive Web App”
is a marketing term

Progressive Web App

Game
Gallery
Book
Progressive Web App

Newspaper
Art Project

Progressive Web Site

Who's behind PWAs?



A Minimum Viable PWA



HTTPS

A Minimum Viable PWA



HTTPS



Web App
Manifest

Web App Manifest

```
{  
  "lang": "en",  
  "short_name": "Wash Post",  
  "name": "The Washington Post",  
  "icons": [ { "src": "img/launcher-icon-2x.png",  
              "sizes": "96x96",  
              "type": "image/png" } ],  
  "start_url": "/pwa/",  
  "display": "standalone",  
  "orientation": "portrait",  
  "background_color": "black"  
}
```

A Minimum Viable PWA



HTTPS



Web App
Manifest



Service
Worker

Should you believe the hype?

A collage of various news articles and video thumbnails related to Progressive Web Apps (PWAs). The headlines include:

- "Why I predict Progressive Web Apps will change the game agencies" by Pratik Dholakiya
- "How Progressive Web Apps Could Change the Mobile Landscape" (Guest Comment, 23 Aug 2018)
- "It's time retailers embrace progressive web apps" (Guest Comment, 23 Aug 2018)
- "circuit breaker: Web apps are only getting better" by Paul Miller (@futurepaul) | Apr 11, 2018, 8:20am EDT

25,962 views | Mar 9, 2018, 07:30am

Why Progressive Web Apps Will Replace Native Mobile Apps

Maybe?

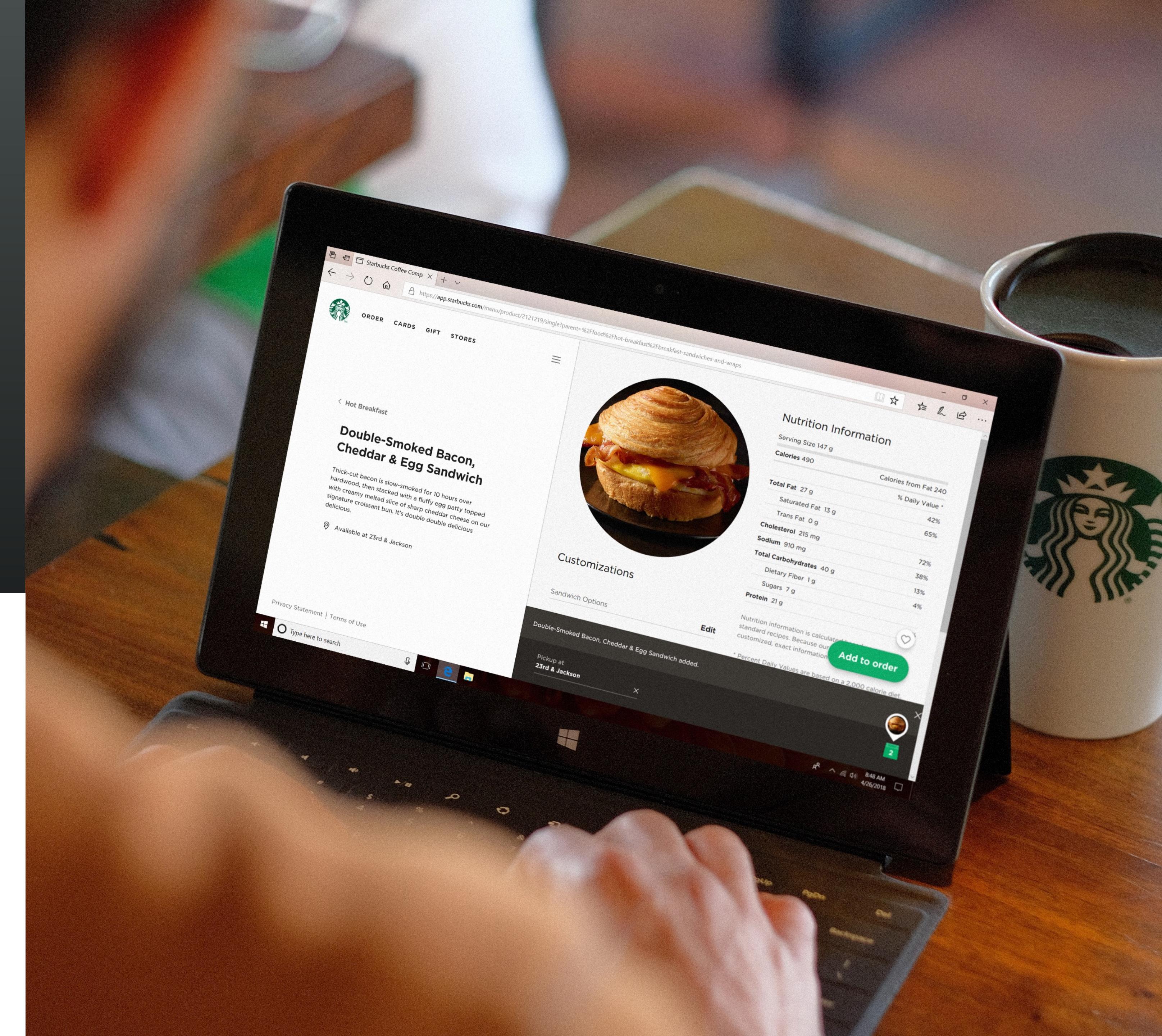
Carnival:
24% opt-in rate and
42% open rate for
push notifications



aka.ms/carnival-pwa



Starbucks: 2x increase in daily active users



aka.ms/google-io-2018

Tinder:
Core experience
with 90% less code



30MB

Android

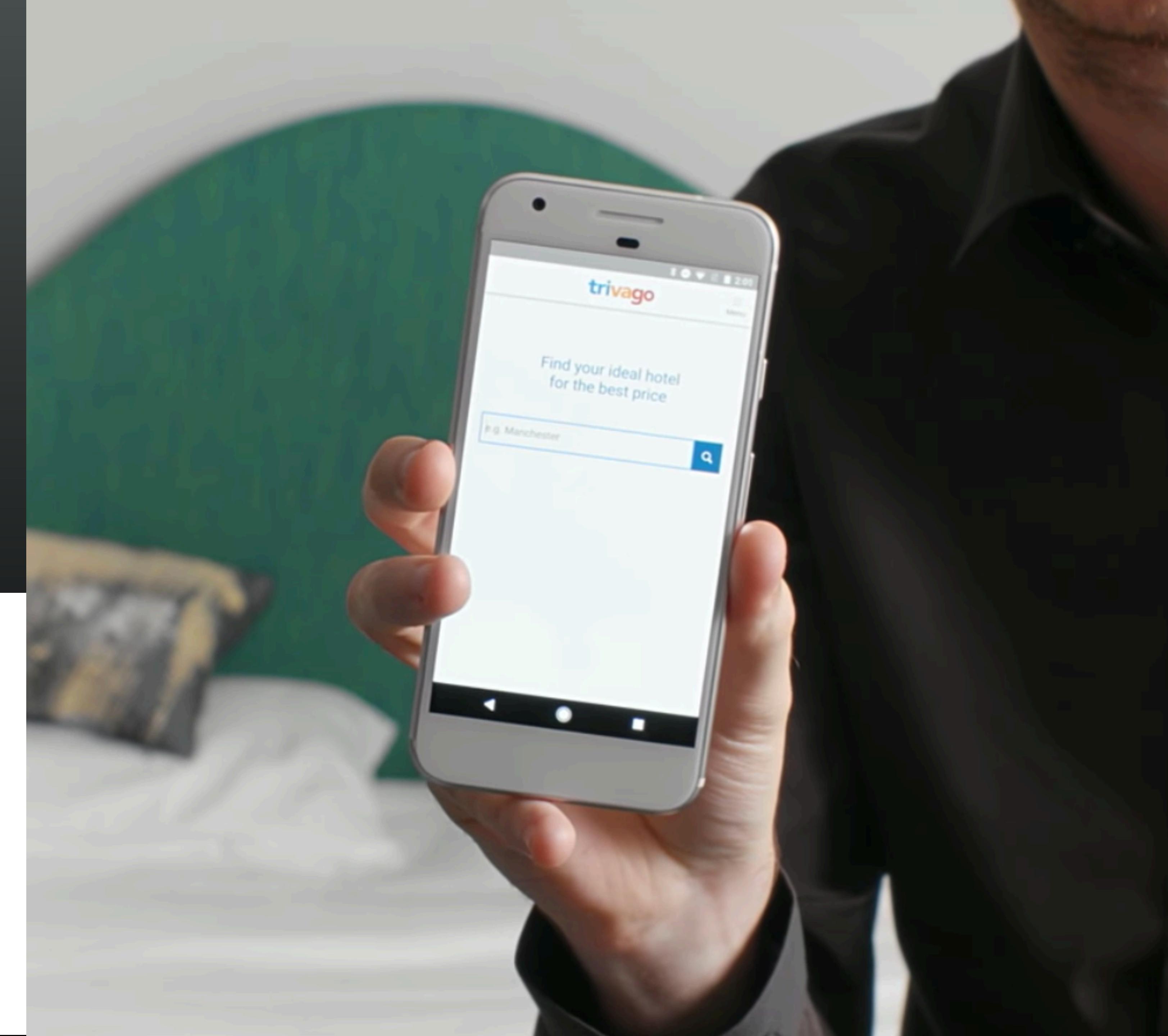
2.8MB

PWA

Trivago:
97% increase in
click-outs to
hotel offers



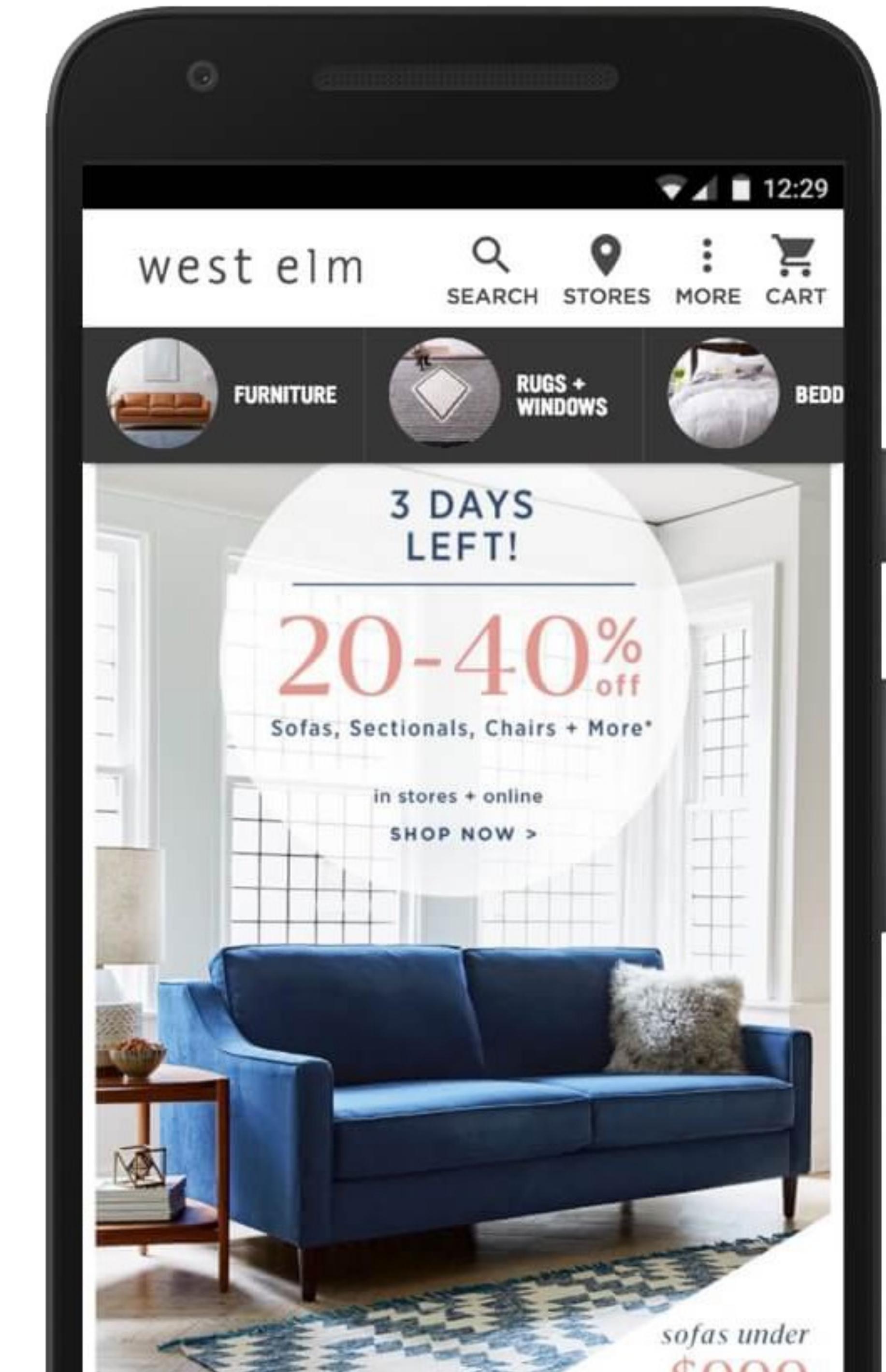
aka.ms/trivago-pwa-2017



West Elm:
15% increase in
time on site
9% increase in
revenue per visit

west elm

aka.ms/west-elm-pwa-2017



PWA Stats

A Minimum Viable PWA



HTTPS

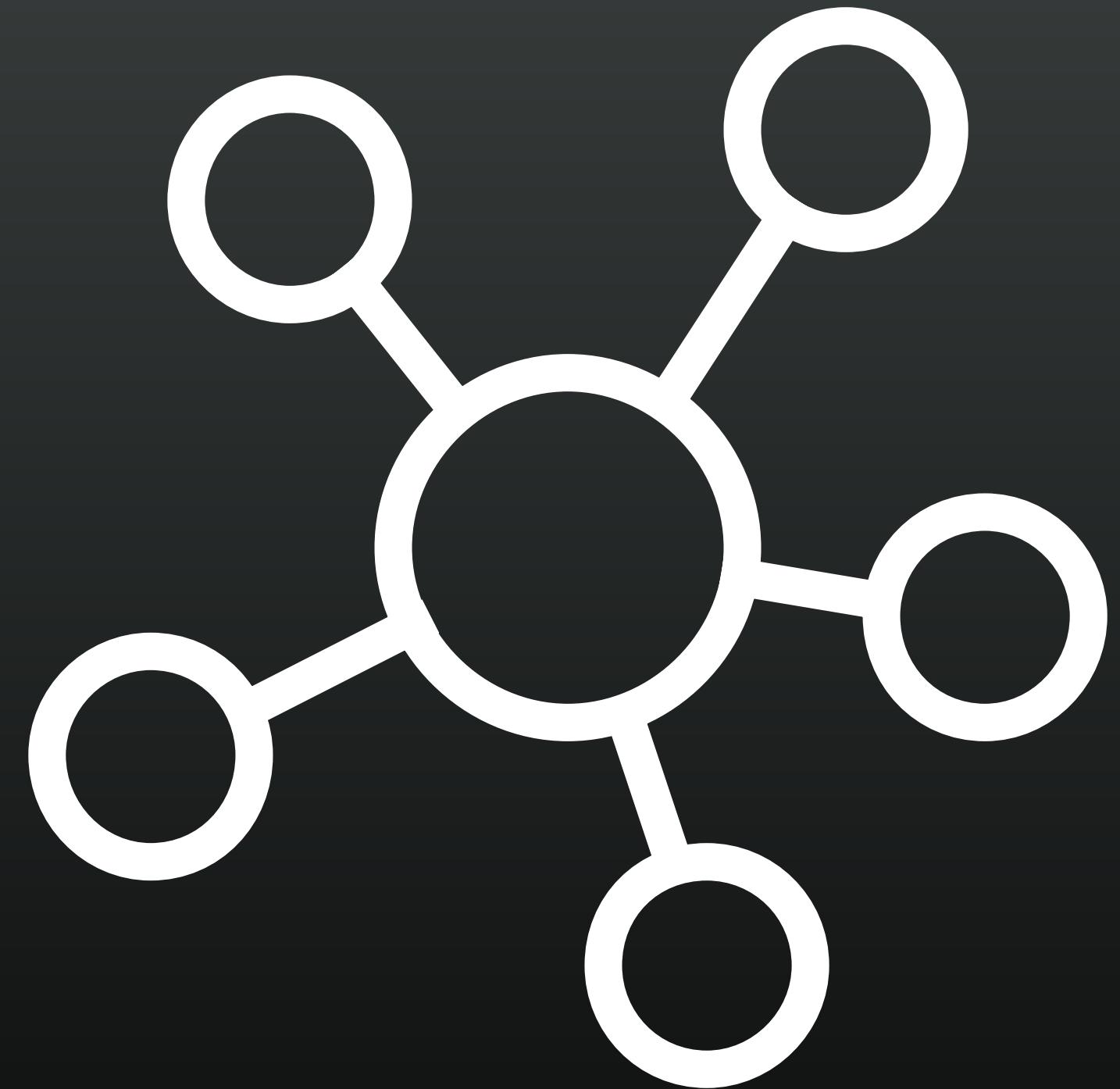


Web App
Manifest



Service
Worker

Let's talk about Service Worker



Registering a Service Worker

```
if ( "serviceworker" in navigator ) {  
  navigator.serviceWorker.register( "/serviceworker.min.js" );  
}  
}
```

Registering a Service Worker

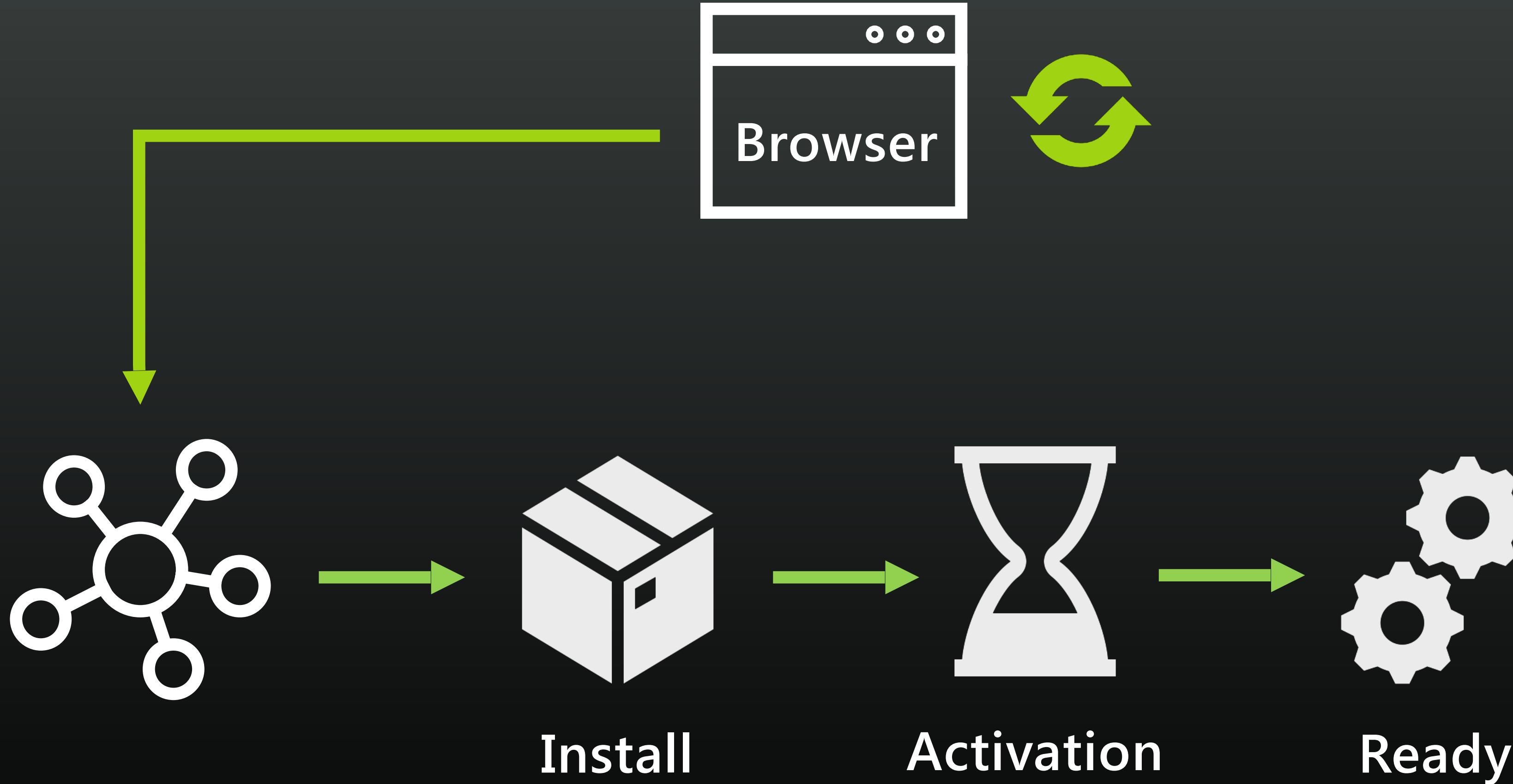
```
if ( "serviceworker" in navigator ) {  
  navigator.serviceWorker.register( "/serviceworker.min.js" );  
}
```

Registering a Service Worker

```
if ( "serviceworker" in navigator ) {  
  navigator.serviceWorker.register( "/serviceworker.min.js" );  
}
```

Path is important!

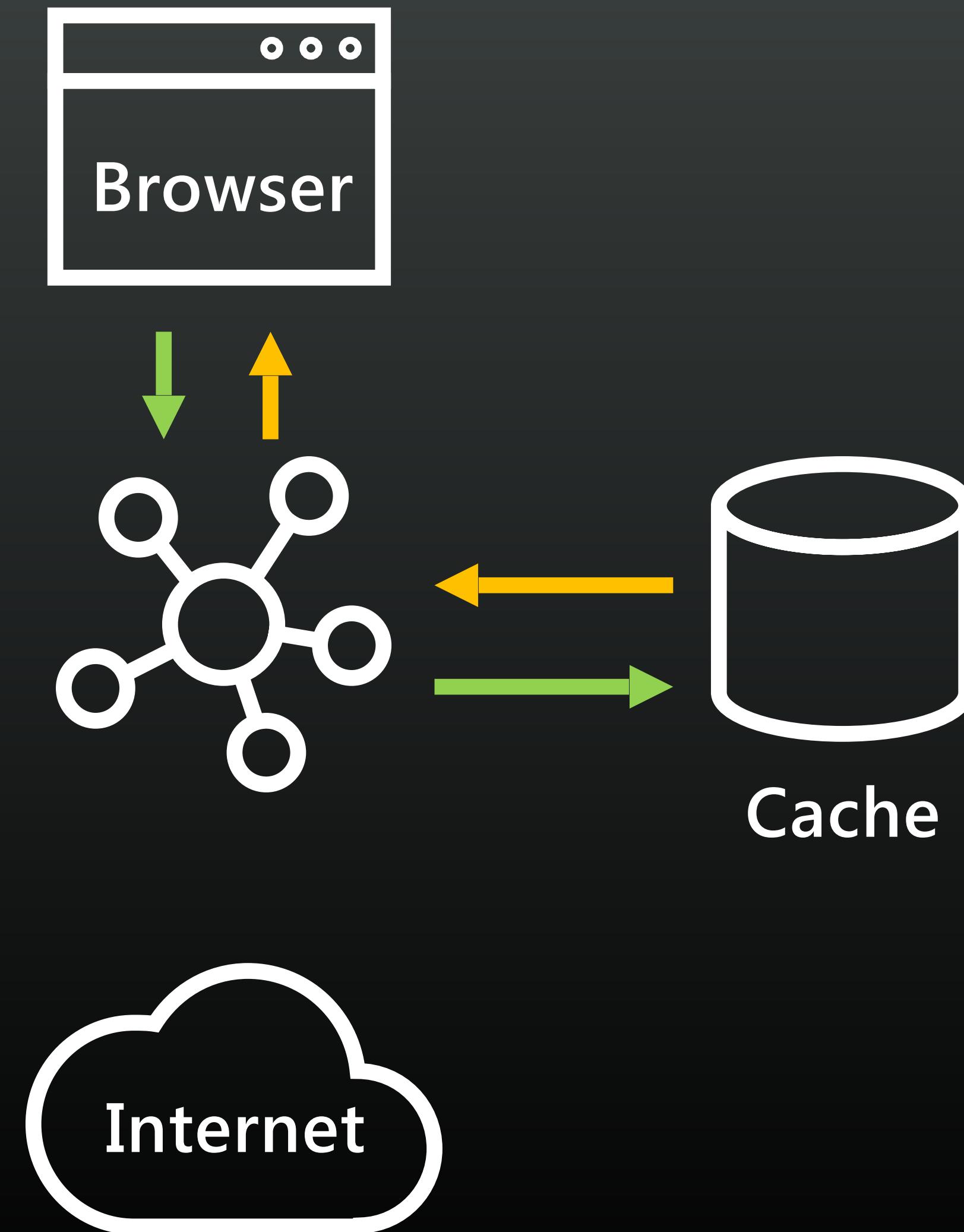
The Service Worker Lifecycle



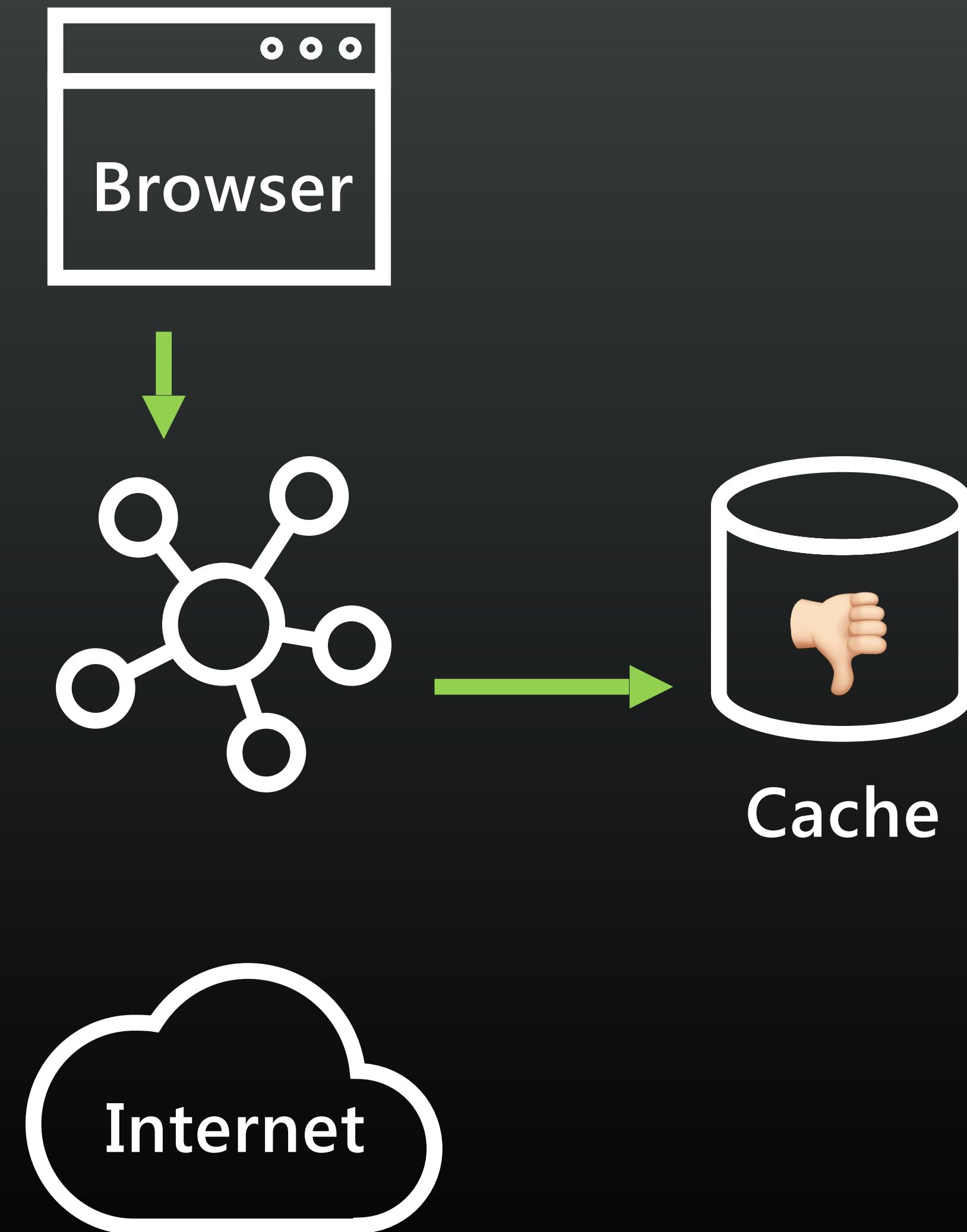
How connections are made



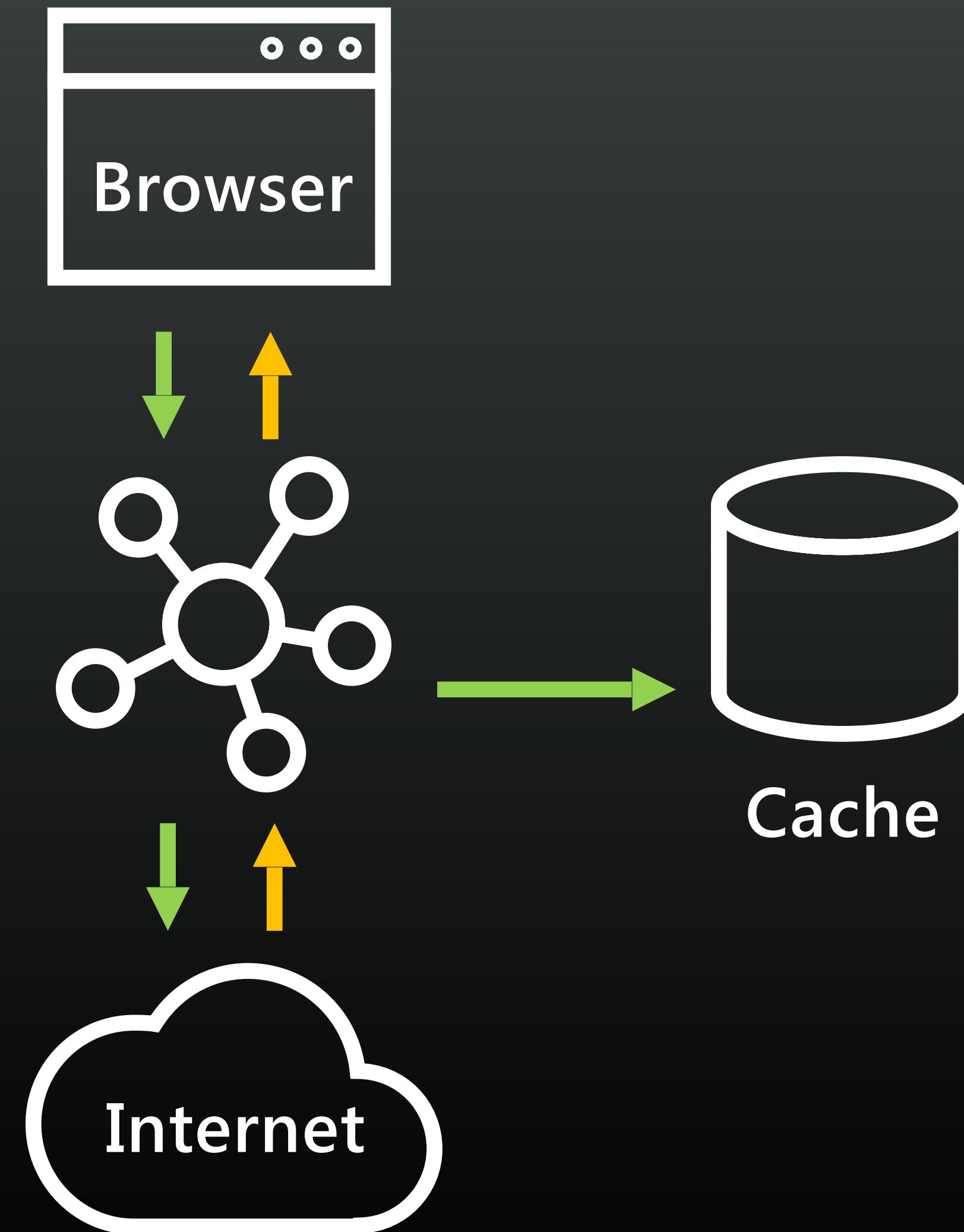
Along comes Service Worker



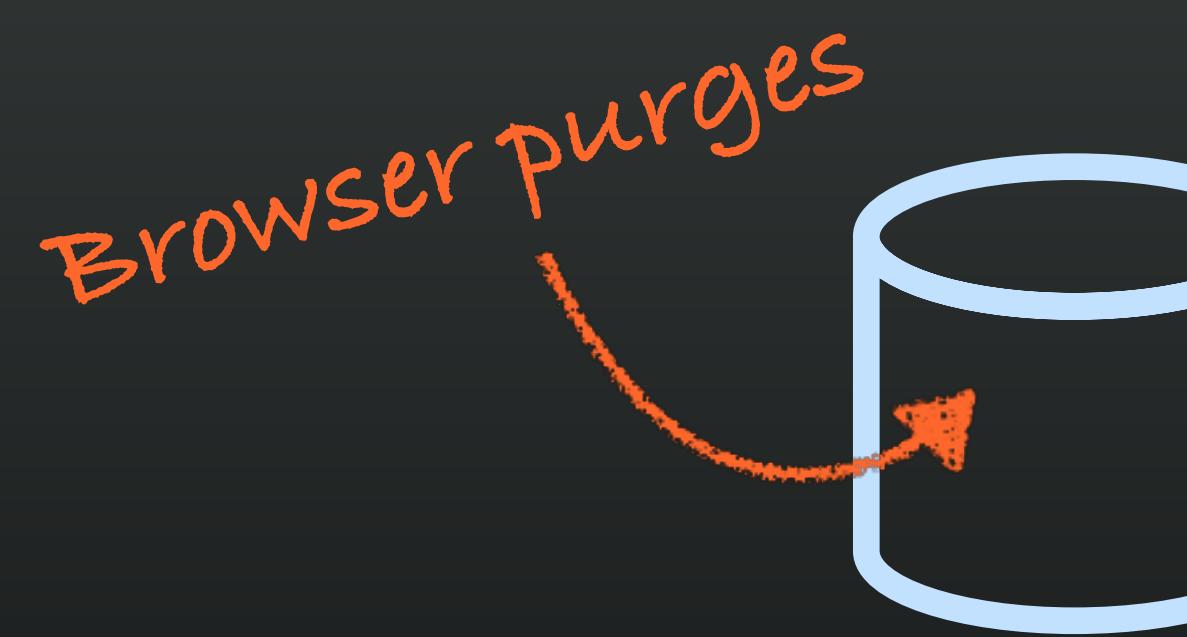
Along comes Service Worker



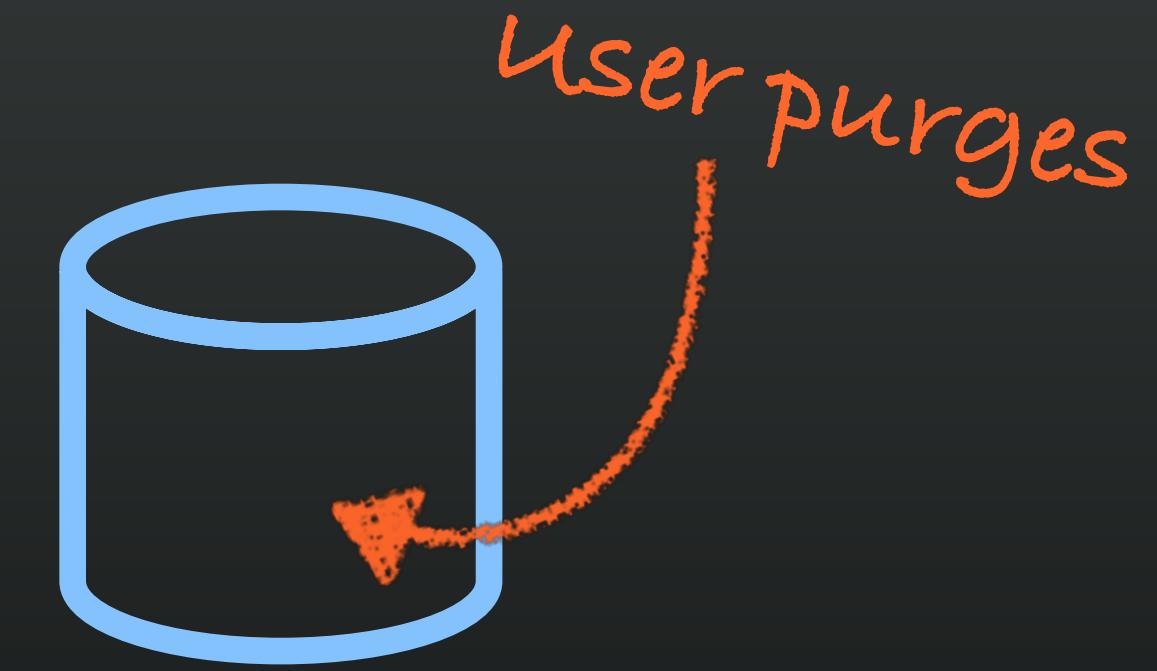
Along comes Service Worker



Know your (storage) limits



Temporary



Persistent

Know your (storage) limits

Volume Size	Domain Limit	Overall Limit
≤ 8 GB		50 MB
8–32 GB	20% of overall	500 MB
32–128 GB		4% of volume
> 128 GB		4% or 20 GB

Except on iOS.
Safari gives you 50 MB.

Raising limits?
Unlimited storage?

Storage is a privilege,
don't abuse it.

How?

#1: No animated GIFs



#2: Use responsive images

```

```

aka.ms/cloudinary-images

#3: Lazy load images

```

```

aka.ms/img-lazy-loading

#4: Provide alternate formats

```
<picture>
  <source type="image/webp" srcset="my.webp">
    
</picture>
```

#4: Provide alternate formats

via Cloudinary URLs:

https://res.cloudinary.com/demo/image/upload/w_300,f_auto/my.jpg

aka.ms/cloudinary-webp

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "install", function( event ){
  event.waitUntil(
    caches.open( "static" ).then( cache => {
      return cache.addAll( ["/i/fallbacks/offline.svg"] );
    })
  );
});
```



```
function respondWithOfflineImage() {
  return caches.match( "/i/fallbacks/offline.svg" );
}
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

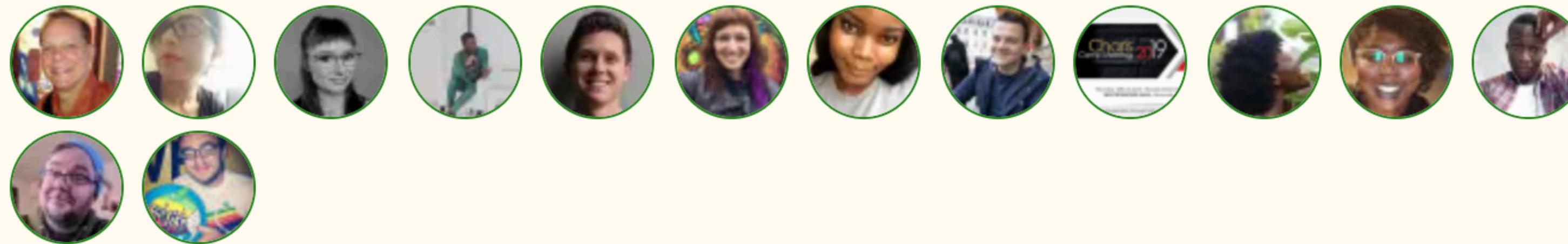
#5: Have fallback images

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      return fetch( request, fetch_config.images )
        .then( response => {
          return response;
        })
        .catch(
          respondWithOfflineImage
        );
    );
  }
});
```

aka.ms/ag-sw

Result!

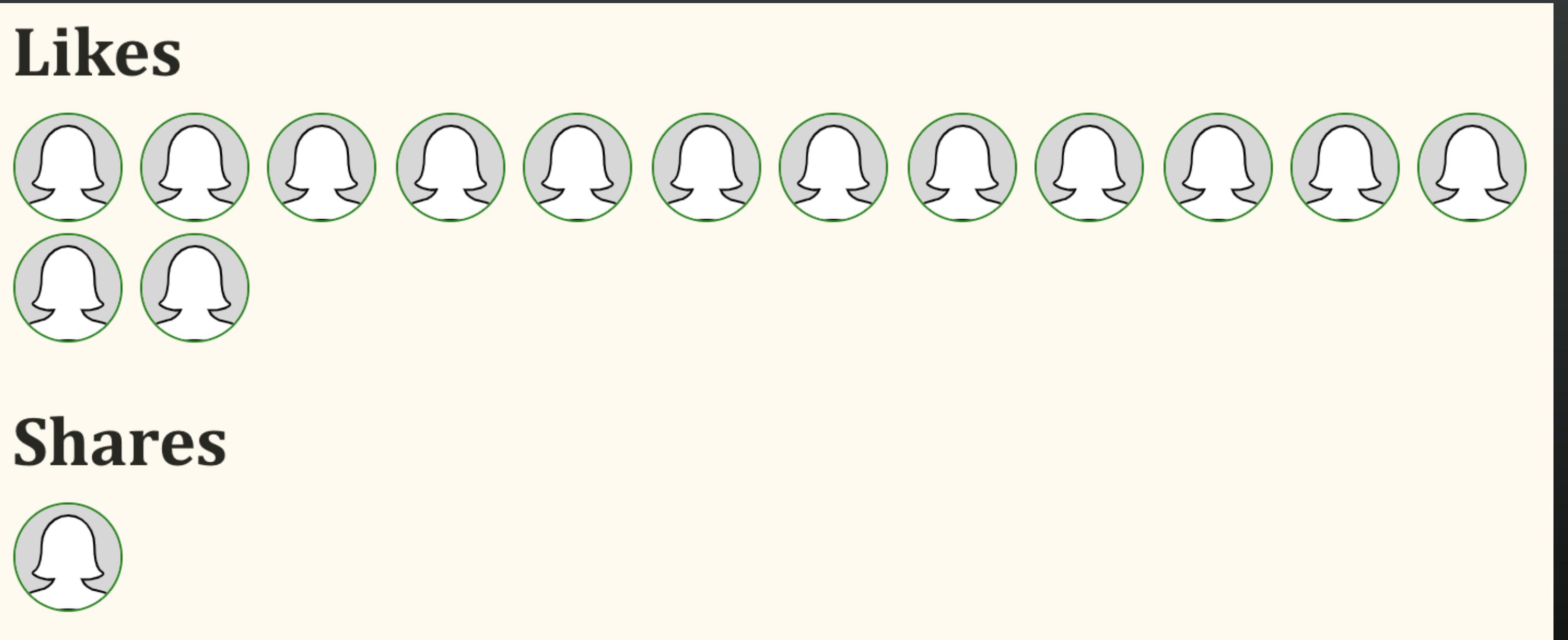
Likes



Shares



Result!



#6: Respect Save Data

```
let save_data = false;  
if ( 'connection' in navigator ) {  
  save_data = navigator.connection.saveData;  
}  
  
// ...  
  
// If save_data is true, the connection will be  
// closed when the page is navigated away from.  
// If save_data is false, the connection will be  
// closed when the page is unloaded.
```

aka.ms/ag-sw

#6: Respect Save Data

```
self.addEventListener( "fetch", event => {
  const request = event.request,
        url = request.url;
  if ( request.headers.get("Accept").includes("image") ) {
    event.respondWith(
      if ( save_data ) {
        return respondWithFallbackImage( url );
      }
      // ...
    );
  }
}
```

aka.ms/ag-sw

#6: Respect Save Data

```
const fallback_avatar = "/i/fallbacks/avatar.svg",
      fallback_image = "/i/fallbacks/image.svg",
      avatars = /webmention\.io/;

// ...

function respondWithFallbackImage( url ) {
  const image = avatars.test( url ) ? fallback_avatar
                                    : fallback_image;
  return caches.match( image );
}
```

aka.ms/ag-sw

#6: Respect Save Data

```
const fallback_avatar = "/i/fallbacks/avatar.svg",
      fallback_image = "/i/fallbacks/image.svg",
      avatars = /webmention\.io/;

// ...

function respondWithFallbackImage( url ) {
  const image = avatars.test( url ) ? fallback_avatar
                                    : fallback_image;
  return caches.match( image );
}
```

aka.ms/ag-sw

#6: Respect Save Data

```
const fallback_avatar = "/i/fallbacks/avatar.svg",
      fallback_image = "/i/fallbacks/image.svg",
      avatars = /webmention\.io/;

// ...

function respondWithFallbackImage( url ) {
  const image = avatars.test( url ) ? fallback_avatar
                                    : fallback_image;
  return caches.match( image );
}
```

aka.ms/ag-sw

Result!



[Emily Grossman](#)

Bingbot is crawling the web for “quality PWAs” to ingest in the Windows Store (aaron-gustafson.com/notebook/progr...).

Here's what Microsoft looks for:



**Data saver active,
media not loaded**

12 October 2017 | [Permalink](#)

#7: Prioritize certain images

```
const high_priority = [
  '/aaron\-.gustafson\.com/',
  '/adaptivewebdesign\.info/'
];

function isHighPriority( url ) {
  let i = high_priority.length;
  while ( i-- ) {
    if ( high_priority[i].test( url ) ) {
      return true;
    }
  }
  return false;
}
```

aka.ms/ag-sw

#7: Prioritize certain images

```
const high_priority = [
  '/aaron\~-gustafson\.com/',
  '/adaptivewebdesign\.info/'
];

function isHighPriority( url ) {
  let i = high_priority.length;
  while ( i-- ) {
    if ( high_priority[i].test( url ) ) {
      return true;
    }
  }
  return false;
}
```

aka.ms/ag-sw

#7: Prioritize certain images

```
const high_priority = [
  '/aaron\~-gustafson\.com/',
  '/adaptivewebdesign\.info/'
];

function isHighPriority( url ) {
  let i = high_priority.length;
  while ( i-- ) {
    if ( high_priority[i].test( url ) ) {
      return true;
    }
  }
  return false;
}
```

aka.ms/ag-sw

#8: Clean up after yourself

```
const version = "v2:",  
  sw_caches = {  
    static: {  
      name: `${version}static`  
    },  
    images: {  
      name: `${version}images`,  
      limit: 75  
    },  
    pages: {  
      name: `${version}pages`,  
      limit: 5  
    },  
    posts: {  
      name: `${version}posts`,  
      limit: 10  
    },  
    other: {  
      name: `${version}other`,  
      limit: 50  
    }  
  };
```

aka.ms/ag-sw

#8: Clean up after yourself

```
function trimCache( cache_name, limit ) {
  caches.open( cache_name ).then( cache => {
    cache.keys().then( items => {
      if ( items.length > limit ) {
        cache.delete( items[0] ).then(
          trimCache( cache_name, limit )
        );
      }
    });
  });
}
```

aka.ms/ag-sw

#8: Clean up after yourself

```
function trimCache( cache_name, limit ) {
  caches.open( cache_name ).then( cache => {
    cache.keys().then( items => {
      if ( items.length > limit ) {
        cache.delete( items[0] ).then(
          trimCache( cache_name, limit )
        );
      }
    });
  });
}
```

aka.ms/ag-sw

#8: Clean up after yourself

```
function trimCache( cache_name, limit ) {
  caches.open( cache_name ).then( cache => {
    cache.keys().then( items => {
      if ( items.length > limit ) {
        cache.delete( items[0] ).then(
          trimCache( cache_name, limit )
        );
      }
    });
  });
}
```

aka.ms/ag-sw

#8: Clean up after yourself

```
if ( "serviceworker" in navigator ) {  
    navigator.serviceWorker.register( "/serviceworker.min.js" );  
  
    if ( navigator.serviceWorker.controller ) {  
        window.addEventListener( "load", function(){  
            navigator.serviceWorker.controller.postMessage( "clean up" );  
        });  
    }  
}
```

aka.ms/ag-sw

#8: Clean up after yourself

```
addEventListener("message", messageEvent => {
  if (messageEvent.data == "clean up") {
    for ( let key in sw_caches ) {
      if ( sw_caches[key].limit != undefined ) {
        trimCache( sw_caches[key].name, sw_caches[key].limit );
      }
    }
  }
});
```

aka.ms/ag-sw

#8: Clean up after yourself

```
addEventListener("message", messageEvent => {
  if (messageEvent.data == "clean up") {
    for ( let key in sw_caches ) {
      if ( sw_caches[key].limit != undefined ) {
        trimCache( sw_caches[key].name, sw_caches[key].limit );
      }
    }
  }
});
```

aka.ms/ag-sw

Make good choices

1. No animated GIFs (especially as backgrounds)
2. Use responsive images
3. Lazy load images
4. Provide alternate image formats
5. Provide fallback images via Service Worker
6. Pay attention to the Save Data header
7. Prioritize certain images
8. Clean up after yourself

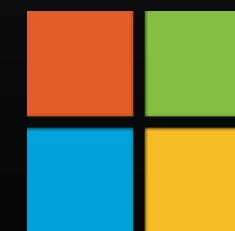


Thank you!

@AaronGustafson

aaron-gustafson.com

slideshare.net/AaronGustafson



Microsoft