

Polymer @ Philly ETE 2016



Jeff Posnick
Google Web DevRel
@jeffposnick



Web Components are **different**.
They are not like web standards
that have come before.

Web Components are low-level **primitives** that let *you* define **your own HTML Elements.**

Template

native client-side templating

Shadow DOM

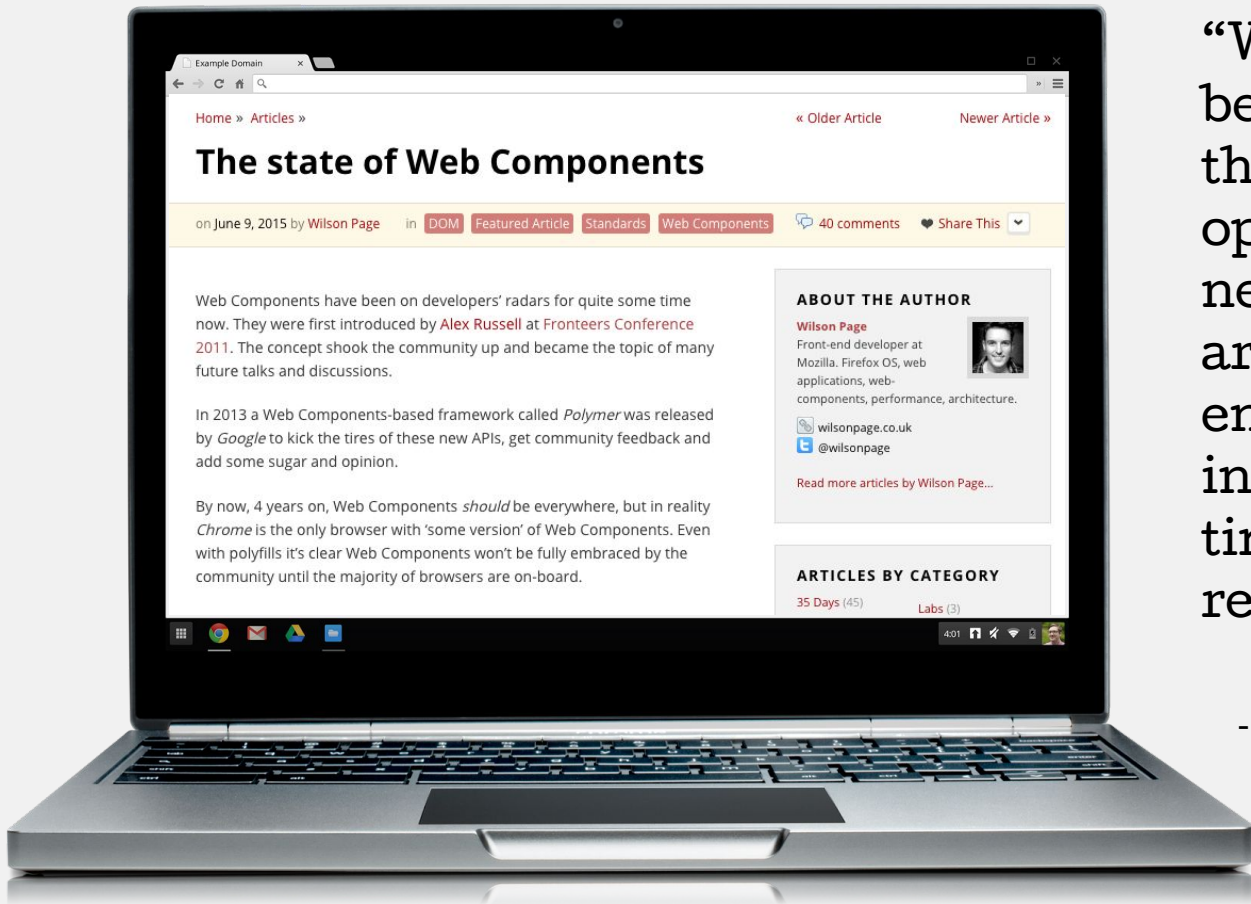
scoping, composition

Custom Elements

define new HTML/DOM

HTML Imports

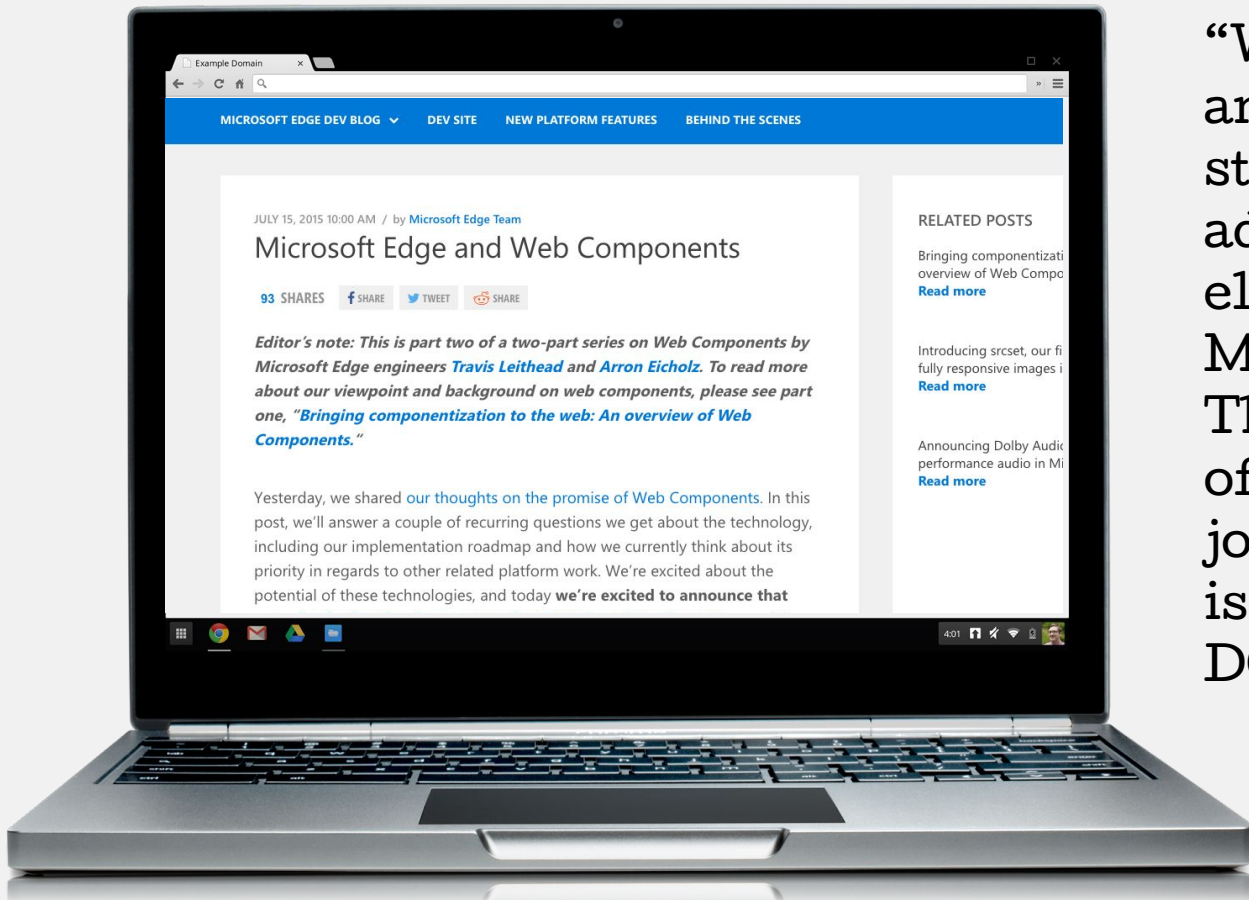
loading web components



“Web Components have been in planning for over three years, but we’re optimistic the end is near. All major vendors are on board, enthusiastic, and investing significant time to help resolve the remaining issues.”

- Wilson Page, Mozilla

Full article: <https://goo.gl/SFaZKS>

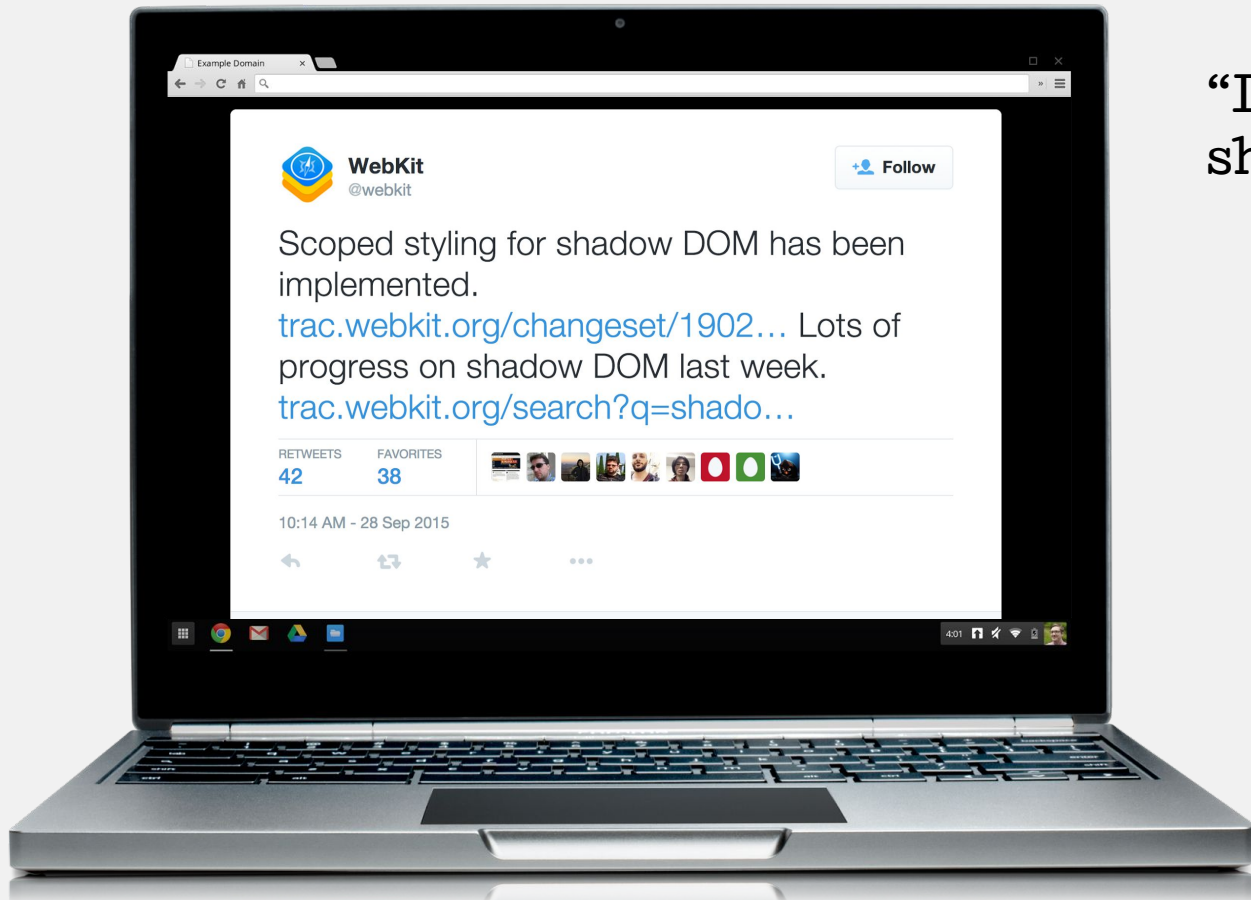


“We are happy to announce that we are starting development to add HTML Template element support to Microsoft Edge today! This is just the beginning of our Web Components journey the next goal is to implement Shadow DOM”

- Travis Leithead & Arron Eicholz,
Microsoft Edge

Full article: <https://goo.gl/N5n4SM>

<https://dev.modern.ie/platform/status/shadowdom/>



“Lots of progress on shadow DOM last week.”

- WebKit Twitter account

<https://twitter.com/webkit/status/648546431771344896>

Primitives are designed so we can
build **libraries on top of them.**

So what is Polymer?

**Polymer is not
a framework**

Salon A - An Introduction to Web Components and Polymer



Jeff Posnick

frameworks

html

polymer

web components



Oops...

Applications

Existing Frameworks

Web Components built with Polymer (or not)

Web Platform

Is it **ready**?



Over 1.3M pages



“We no longer build applications. We have, and are enriching, a module market sourced from industry and the ING global community. Modules are assembled into applications as the business requires.”

- Ben Issa, ING Head of IT Strategy

Full article: <http://goo.gl/gJbdKu>

A thin, light-colored line graph showing an upward trend on a dark blue background. The line starts at a low point on the left and rises steadily towards the top right, with several small curves and inflection points, suggesting a non-linear growth pattern.

300+ Google projects



Ok Google, play music

Free, curated radio for what you're doing, how you feel, or what you want to hear.

[LISTEN NOW](#)



JOIN IN AND LEVEL UP

SUMMER 2015



Be one of the first to know when
YouTube Gaming goes live.

[GET NOTIFIED](#)

[LEARN MORE](#)



The background features three large, overlapping geometric shapes: a blue one on the left, a pink one in the center, and a purple one on the right. Each shape is composed of several triangular and quadrilateral facets, creating a faceted, crystalline appearance. A small, solid blue circle is positioned at the bottom center of the composition.

**READY FOR
PRODUCTION**

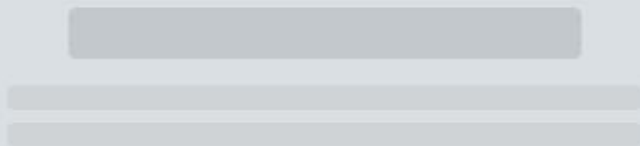


**Let's build
an element!**



http://www.webaddress.com

Hey user! Something **awesome** happened!



Hey user! Something **awesome** happened!



`<alert-banner>`

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
<dom-module id="alert-banner">
  <template>
  </template>
  <script>
    Polymer({
      is: 'alert-banner'
    });
  </script>
</dom-module>
```


alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
```

```
<dom-module id="alert-banner">
```

```
  <template>
```

```
</template>
```

```
  <script>
```

```
    Polymer({
```

```
      is: 'alert-banner'
```

```
    });
```

```
  </script>
```

```
</dom-module>
```



**Import all of your
*dependencies***

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
```

```
<dom-module id="alert-banner">
```

```
  <template>
```

```
  </template>
```

```
  <script>
```

```
    Polymer({
```

```
      is: 'alert-banner'
```

```
    });
```

```
  </script>
```

```
</dom-module>
```



A container for your
element definition

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
```

```
<dom-module id="alert-banner">
```

```
  <template>
```

```
  </template>
```

```
  <script>
```

```
    Polymer({
```

```
      is: 'alert-banner'
```

```
    });
```

```
  </script>
```

```
</dom-module>
```



Local DOM is the DOM
an elements is *in charge of*
creating and managing

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
<dom-module id="alert-banner">
  <template>
    <style>
      .alert { background: blue; color: white; }
    </style>
    <div class="alert">
      Hey user! Something awesome happened!
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner'
    });
  </script>
</dom-module>
```



A screenshot of a web browser window. The address bar shows "http://www.webaddress.com". A bright blue banner is displayed at the top of the page with the text "Hey user! Something awesome happened!". A white arrow points from the code on the left to the banner in the browser.

Hey user! Something **awesome** happened!

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">

<dom-module id="alert-banner">
  <template>
    <style>
      .alert { background: blue; color: white; }
    </style>
    <div class="alert">
      Hey user! Something awesome happened!
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner'
    });
  </script>
</dom-module>
```

Define your
prototype



</header>

</nav>

<main>

<h1 class="logo">Polymer!</h1>

<alert-banner></alert-banner>

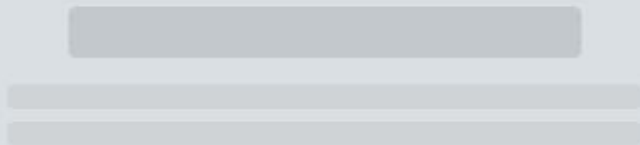
<section>

<article class="top-story">



http://www.webaddress.com

Hey user! Something **awesome** happened!



alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">

<dom-module id="alert-banner">
  <template>
    <style>
      .alert { background: blue; color: white; }
    </style>
    <div class="alert">
      Hey user! Something awesome happened!
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner'
    });
  </script>
</dom-module>
```

Replace hard-coded data



alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">

<dom-module id="alert-banner">
  <template>
    <style>
      .alert { background: blue; color: white; }
    </style>
    <div class="alert">
      <content select=".message"></content>
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner'
    });
  </script>
</dom-module>
```



With content elements!

Light DOM - The world outside
your component's **Local DOM**

alert-banner.html

```
<link rel="import" href="../../polymer/polymer.html">
```

```
<dom-module id="alert-banner">
```

```
  <template>
```

```
    <style>
```

```
    .alert { background: blue; color: white; }
```

```
  </style>
```

```
  <div class="alert">
```

```
    <content select=".message"></content>
```

```
  </div>
```

```
</template>
```

```
<script>
```

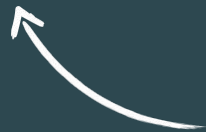
```
  Polymer({
```

```
    is: 'alert-banner'
```

```
  });
```

```
</script>
```

```
</dom-module>
```



Select content with
CSS selectors

```
</header>
</nav>
<main>
  <h1 class="logo">Polymer!</h1>
  <alert-banner></alert-banner>
  <section>
    <article class="top-story">
      
      <p>Lorem ipsum dolor sit amet...</p>
    </article>
```

```
</header>
</nav>
<main>
  <h1 class="logo">Polymer!</h1>
  <alert-banner>
    <span class="message">
      Success! Your first component!
    </span>
  </alert-banner>
</section>
```

```
</header>
</nav>
<main>
  <h1 class="logo">Polymer!</h1>
  <alert-banner>
    <span class="message">
```

Matching class



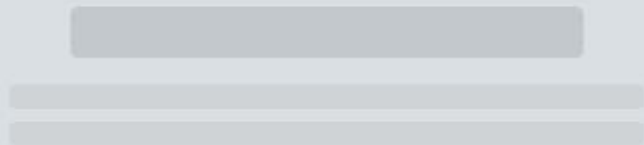
Success! Your first component!

```
</span>
</alert-banner>
<section>
```



http://www.webaddress.com

Success! Your first component!



```
<input type="checkbox">
```




```
<input type="checkbox" checked>
```



```
<input type="checkbox" checked>
```

```
// or in JavaScript  
input.checked = true;
```




```
<dom-module id="alert-banner">
  <template>
    ...
    <div class="alert">
      <content select=".message"></content>
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner',
      properties: {
        shown: {
          type: Boolean,
          value: false,
          notify: true
        }
      }
    })
  </script>

```

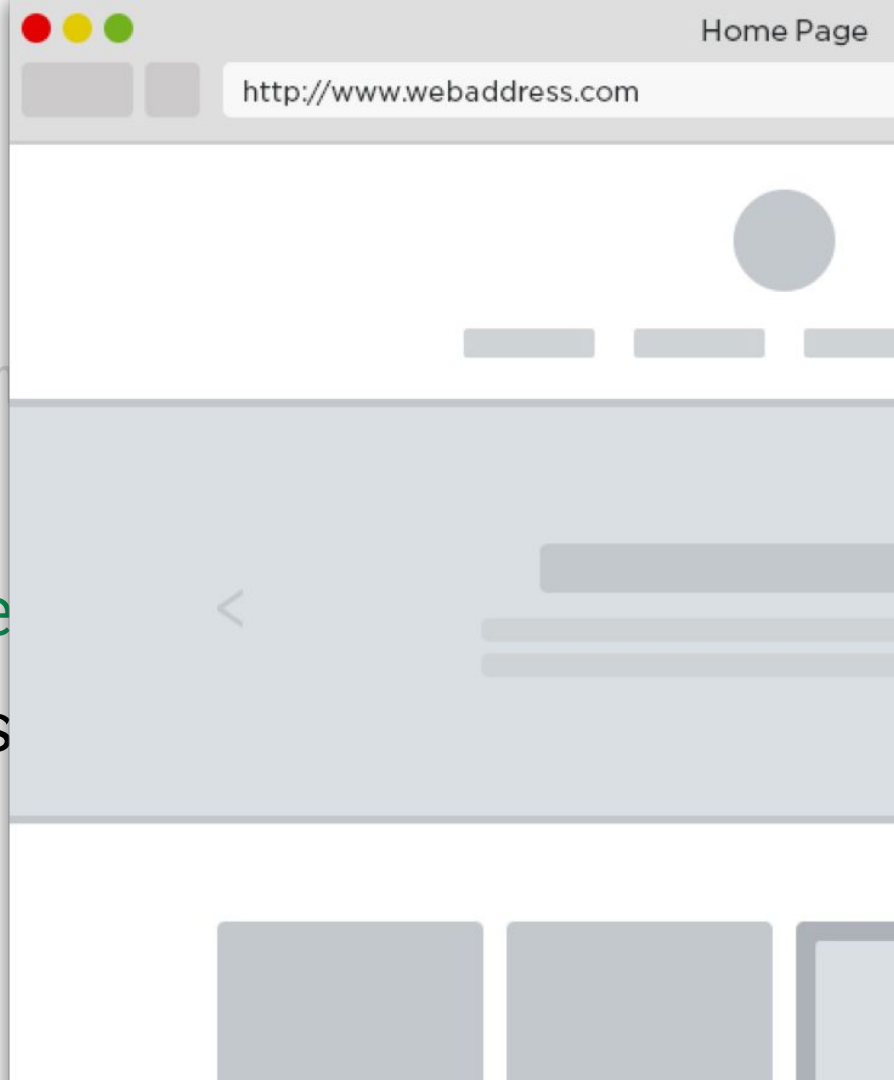
← Properties make your
element ***configurable***

```
<dom-module id="alert-banner">
  <template>
    ...
    <div class="alert" hidden$="{{!shown}}">
      <content select=".message"></content>
    </div>
  </template>
  <script>
    Polymer({
      is: 'alert-banner',
      properties: {
        shown: {
          type: Boolean,
          value: false,
          notify: true
        }
      }
    })
  </script>
</dom-module>
```

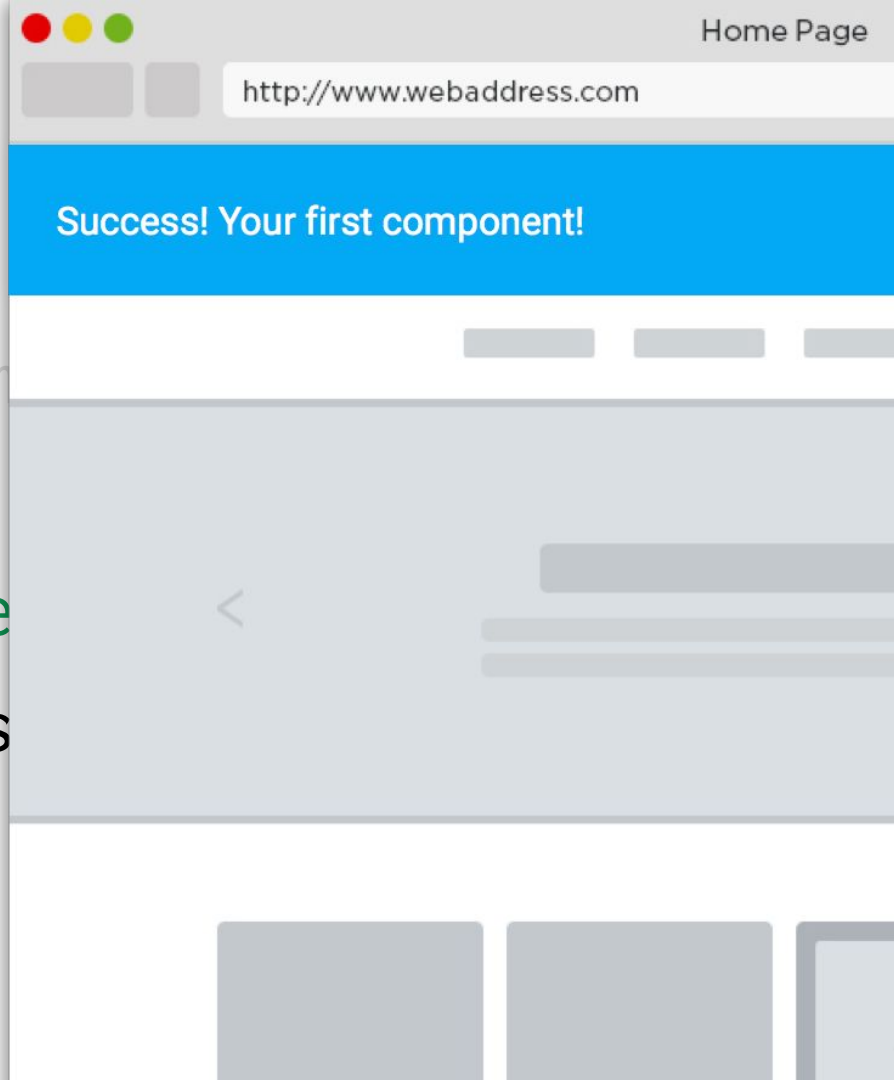
Bind attributes to the
state of a property



```
</header>
</nav>
<main>
  <h1 class="logo">Polym
  <alert-banner>
    <span class="message"
      Success! Your first
    </span>
  </alert-banner>
</section>
```



```
</header>
</nav>
<main>
  <h1 class="logo">Polym
  <alert-banner shown>
    <span class="message"
      Success! Your first
    </span>
  </alert-banner>
</section>
```



Elements

Building blocks for a better web

“There’s an element for that!”

1.0

Fe

**Iron
Elements**

Polymer core elements

1.0

Md

**Paper
Elements**

UI components following the
material design guidelines

1.0

Fe

Iron
Elements

Polymer core elements

1.0

Md

Paper
Elements

UI components following the
material design guidelines

1.0

Go

Google Web
Components

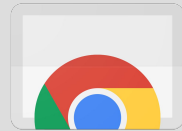
Components for Google's APIs
and services

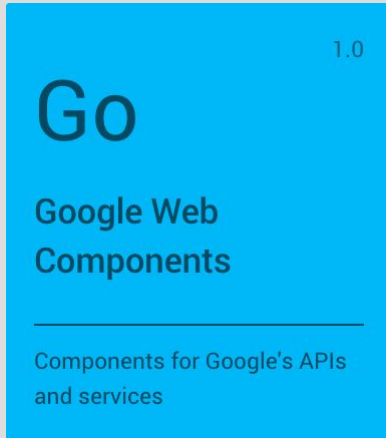
1.0

Go

Google Web Components

Components for Google's APIs and services





<google-drive>

<google-analytics>

<google-calendar>

<google-map>

<google-hangout-button>

<google-sheets>

<google-cloud>

<google-translate>

<google-castable-video>

<google-youtube>

1.0

Fe

**Iron
Elements**

Polymer core elements

1.0

Md

**Paper
Elements**

UI components following the
material design guidelines

1.0

Go

**Google Web
Components**

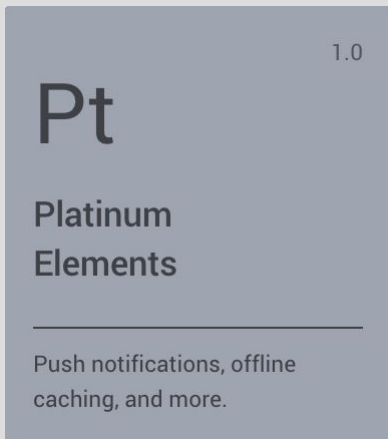
Components for Google's APIs
and services

1.0

Pt

**Platinum
Elements**

Push notifications, offline
caching, and more.



<platinum-sw>

<platinum-sw-cache>

<platinum-sw-register>

<platinum-sw-fetch>

<platinum-push-messaging>

...

1.0

Fe

Iron Elements

Polymer core elements

1.0

Md

Paper Elements

UI components following the
material design guidelines

1.0

Go

Google Web Components

Components for Google's APIs
and services

1.0

Pt

Platinum Elements

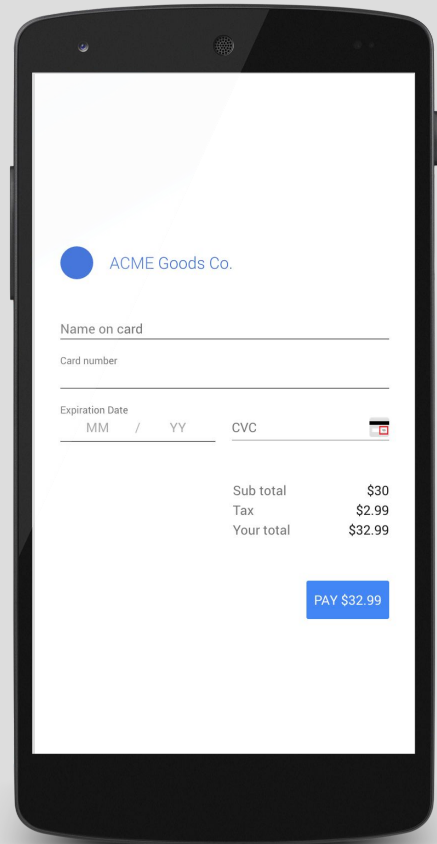
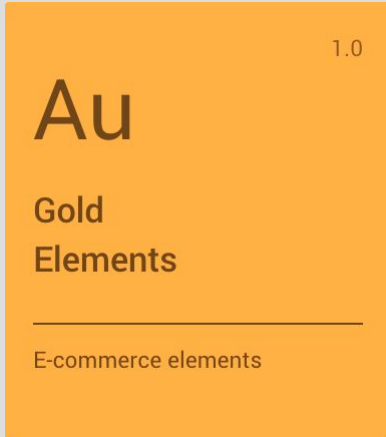
Push notifications, offline
caching, and more.

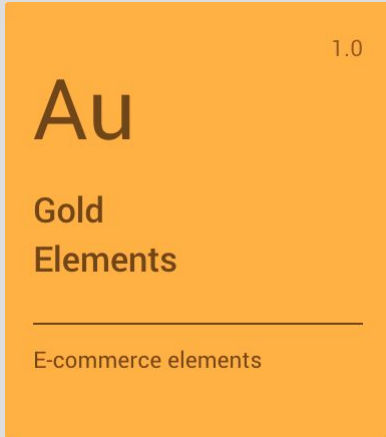
1.0

Au

Gold Elements

E-commerce elements





<gold-cc-input>

<gold-cc-expiration-input>

<gold-cc-cvc-input>

<gold-zip-input>

<gold-phone-input>

<gold-email-input>

...



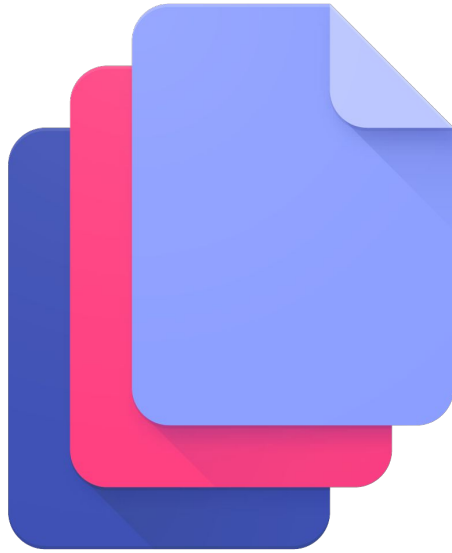
<p>1.0</p> <h1>Fe</h1> <p>Iron Elements</p> <hr/> <p>Polymer core elements</p>	<p>1.0</p> <h1>Md</h1> <p>Paper Elements</p> <hr/> <p>UI components following the material design guidelines</p>	<p>1.0</p> <h1>Go</h1> <p>Google Web Components</p> <hr/> <p>Components for Google's APIs and services</p>
<p>1.0</p> <h1>Pt</h1> <p>Platinum Elements</p> <hr/> <p>Push notifications, offline caching, and more.</p>	<p>1.0</p> <h1>Au</h1> <p>Gold Elements</p> <hr/> <p>E-commerce elements</p>	

elements.polymer-project.org

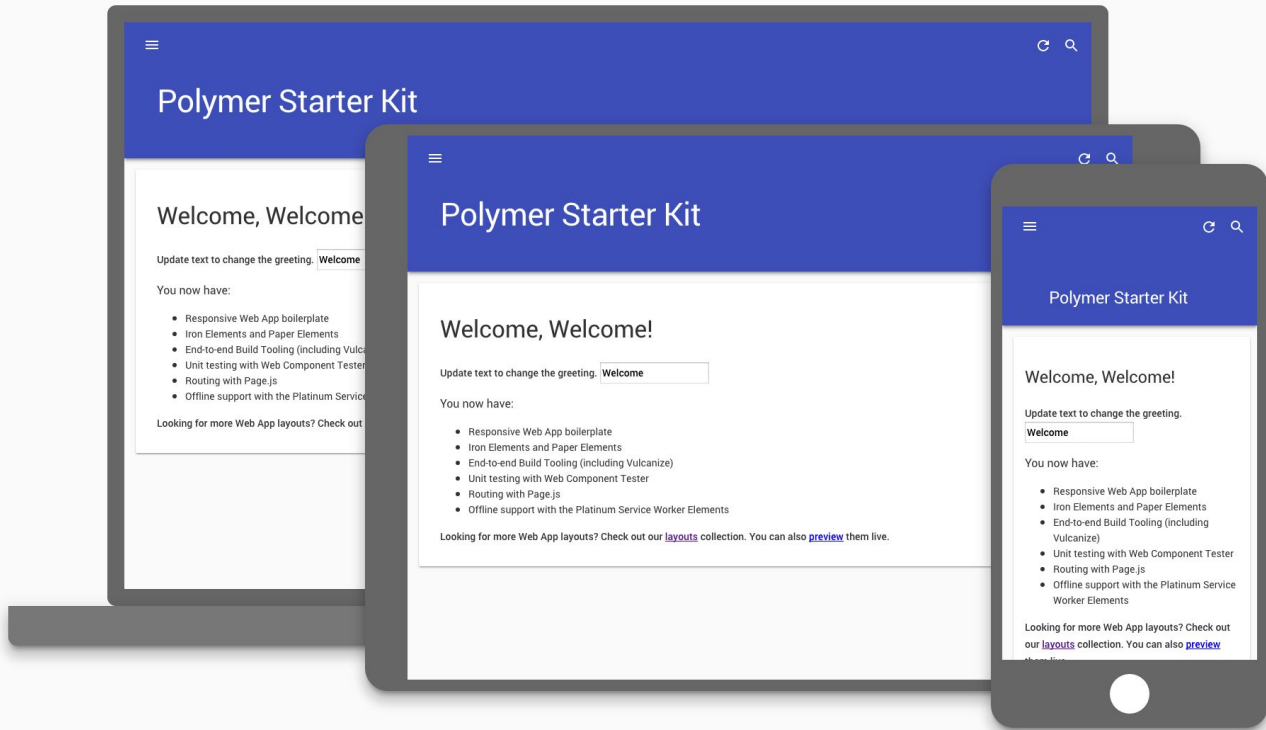
“There’s an element for that!”

Applications

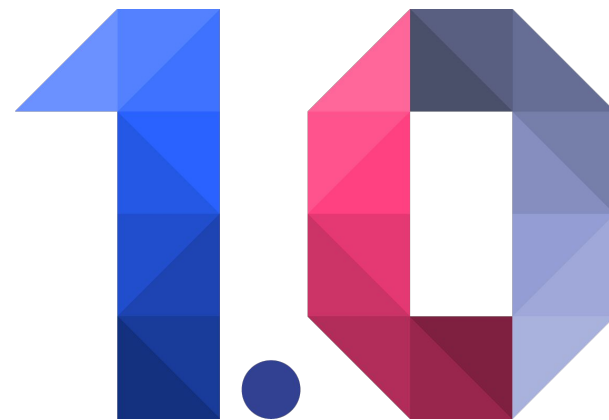
Combining elements into something great



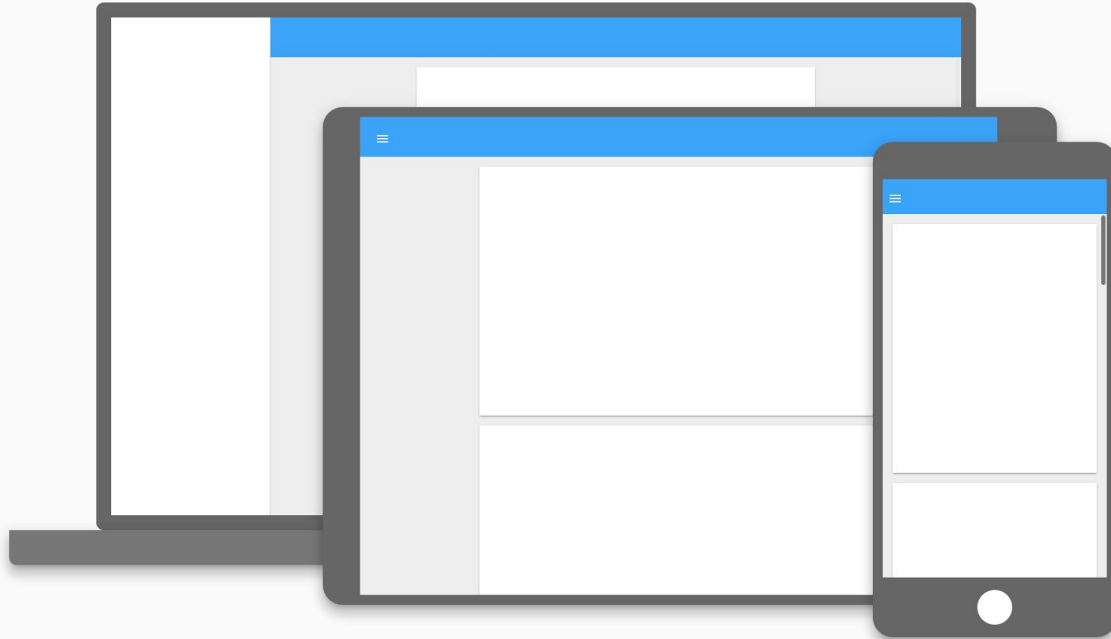
Polymer Starter Kit



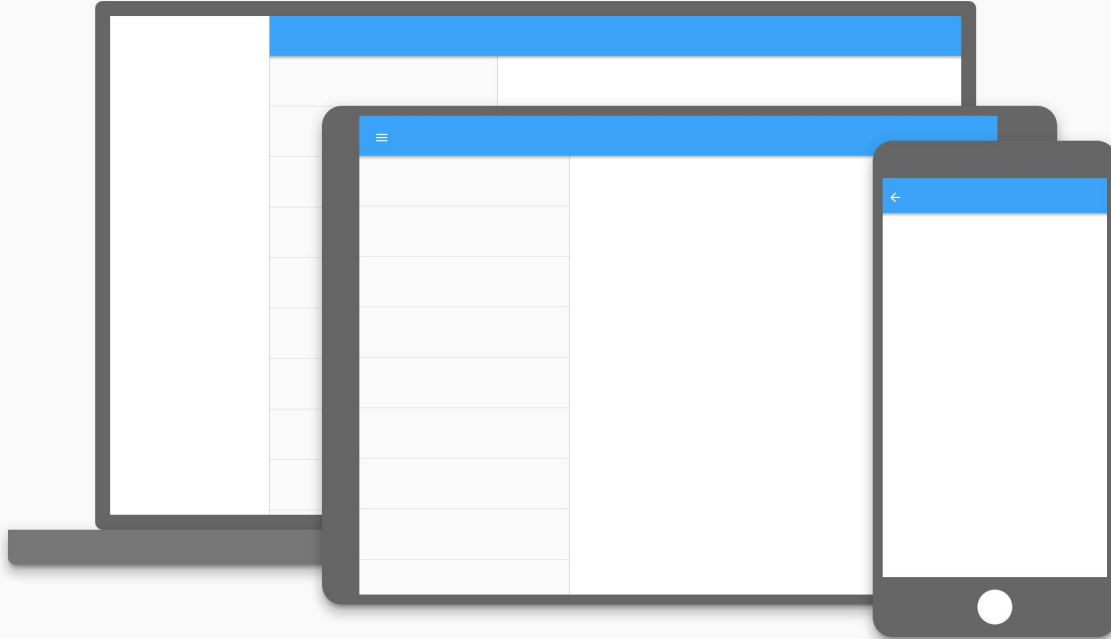
Polymer Starter Kit



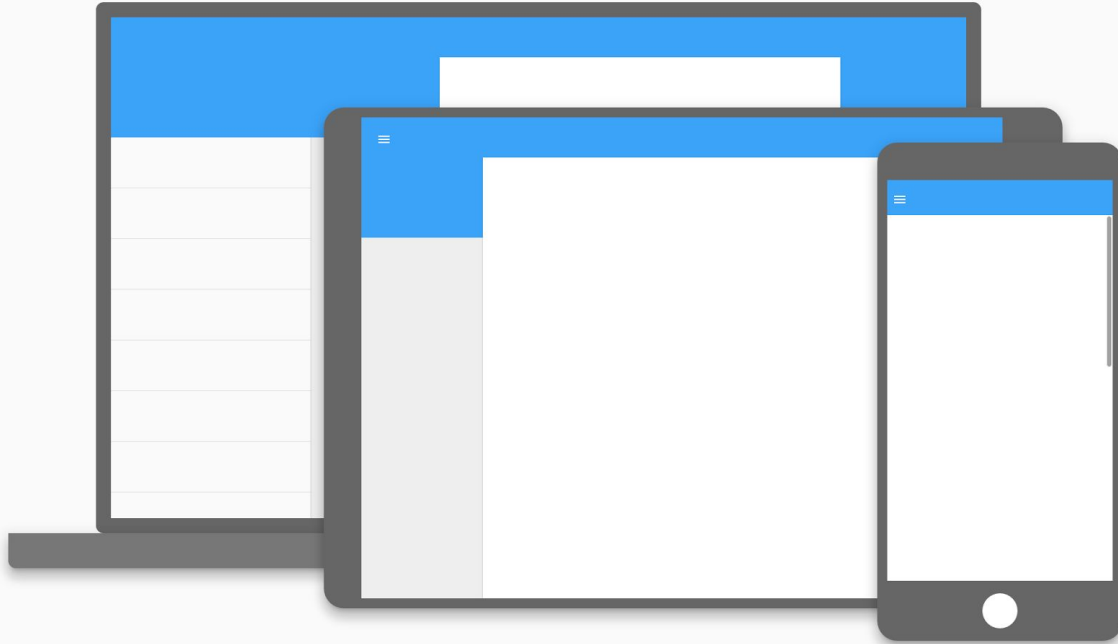
App templates



Navigation Cards Layout



Navigation List Detail

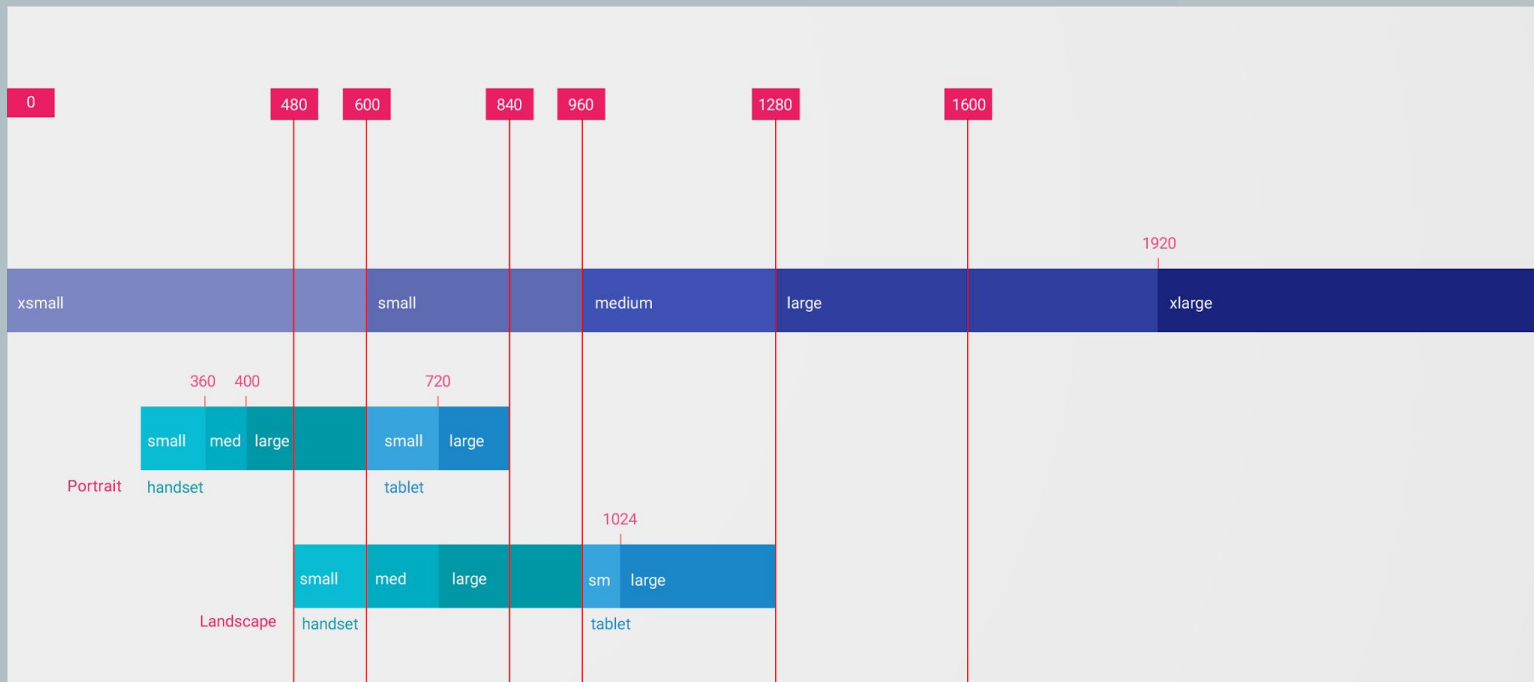


List Card Over



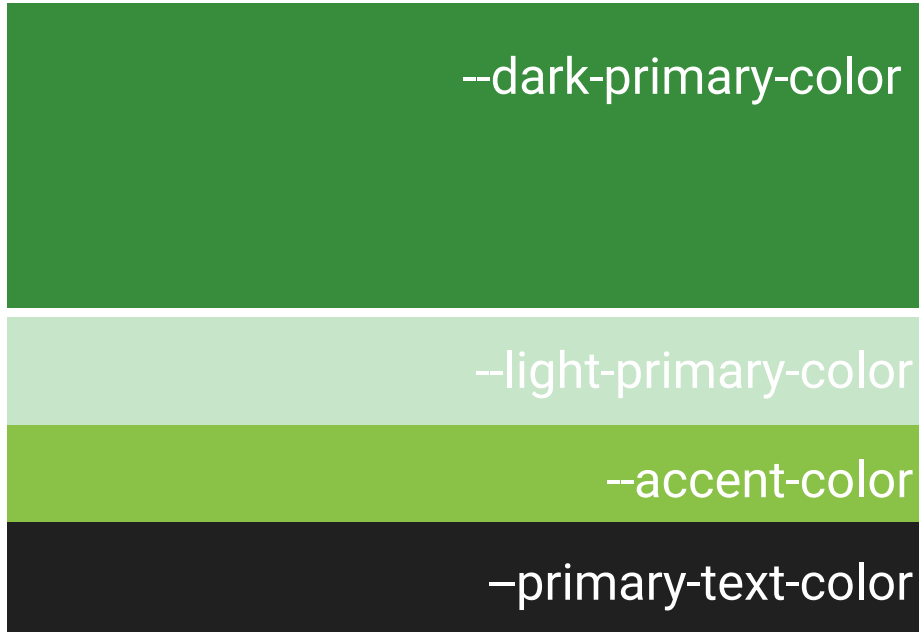
Responsive breakpoints



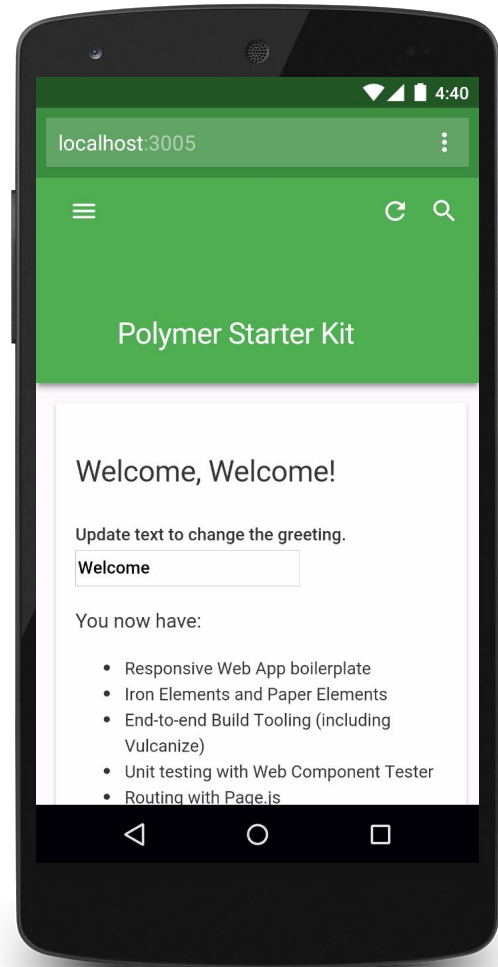


Material Design breakpoints

App-wide Theming



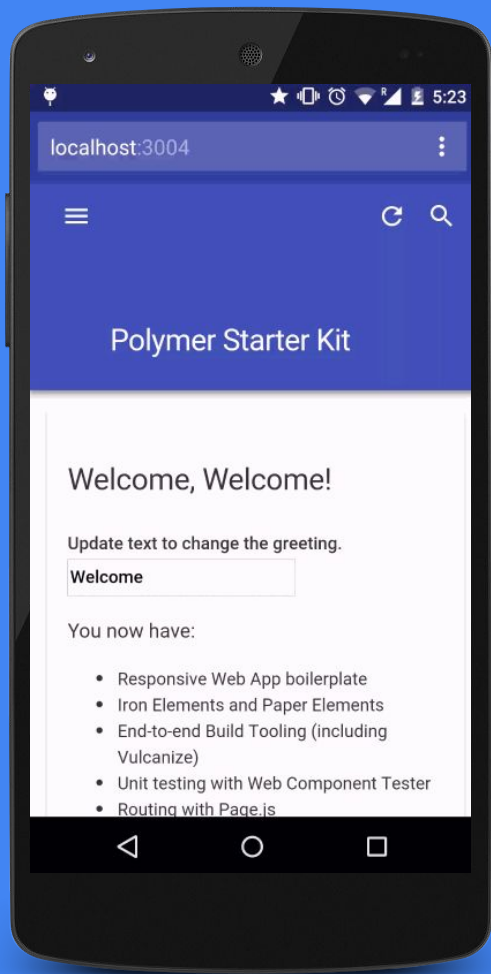
CSS custom properties





Offline-first (in your app)





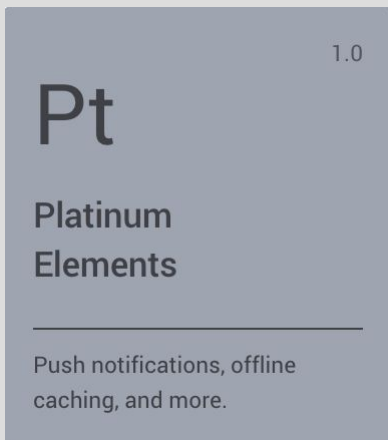

```
// updated service worker is activated.
var CACHE_VERSION = 1;
var CURRENT_CACHES = {
  'read-through': 'read-through-cache-v' + CACHE_VERSION
};

self.addEventListener('activate', function(event) {
  // Delete all caches that aren't named in CURRENT_CACHES.
  // While there is only one cache in this example, the same logic will handle the case where
  // there are multiple versioned caches.
  var expectedCacheNames = Object.keys(CURRENT_CACHES).map(function(key) {
    return CURRENT_CACHES[key];
  });

  event.waitUntil(
    caches.keys().then(function(cacheNames) {
      return Promise.all(
        cacheNames.map(function(cacheName) {
          if (expectedCacheNames.indexOf(cacheName) == -1) {
            // If this cache name isn't present in the array of "expected" cache names, then delete it.
            console.log('Deleting out of date cache:', cacheName);
            return caches.delete(cacheName);
          }
        })
      );
    })
  );
});
```

```
// This sample illustrates an aggressive approach to caching, in which every valid response is
// cached and every request is first checked against the cache.
// This may not be an appropriate approach if your web application makes requests for
// arbitrary URLs as part of its normal operation (e.g. a RSS client or a news aggregator),
// as the cache could end up containing large responses that might not end up ever being accessed.
// Other approaches, like selectively caching based on response headers or only caching
// responses served from a specific domain, might be more appropriate for those use cases.
self.addEventListener('fetch', function(event) {
  console.log('Handling fetch event for', event.request.url);
```

“There’s an element for that!”



<platinum-sw>

<platinum-sw-cache>

<platinum-sw-register>

<platinum-sw-fetch>

<platinum-push-messaging>

...

Service Worker caching

Take your app offline with ease

```
<link rel="import" href="platinum-sw-elements.html">
```

```
<platinum-sw>
```

```
  <platinum-sw-cache default-cache-strategy="networkFirst"  
    precache='["image.jpg", "results.json", "page.html"]'>
```

```
  </platinum-sw-cache>
```

```
</platinum-sw>
```

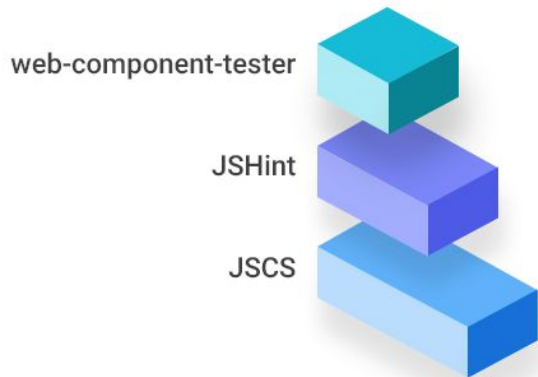
Production-ising Apps



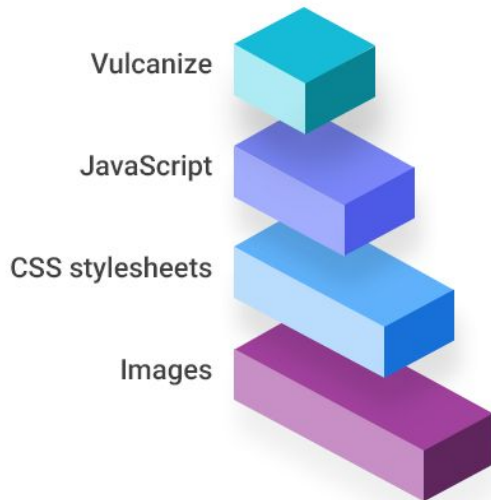
Build process out of the box



QUALITY & TESTING



MINIFICATION

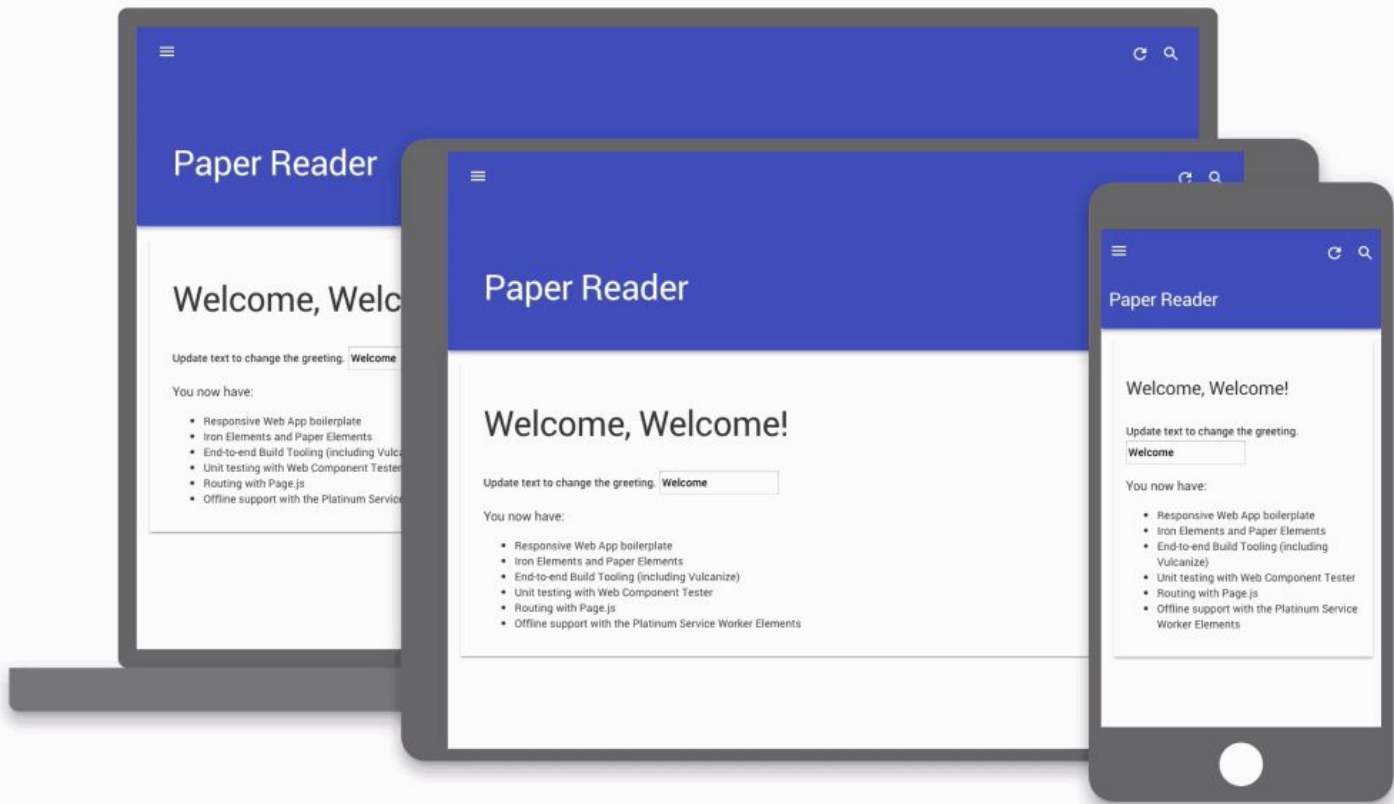


OFFLINE



CROSS-DEVICE RELOAD





Live-reload and cross-device development

DOWNLOAD

developers.google.com/web/tools/polymer-starter-kit

Polymer is lean, and production ready

Create or reuse elements in any app

Starter Kit gets you up and running

Web Components polyfills

Polymer!

An HTML file

bit.ly/polycasts

polymer-project.org

Thank You!



Jeff Posnick
Google Web DevRel
@jeffposnick