

# Working Example - Structure



Bill Stavroulakis

@bstavroulakis | [bstavroulakis.com](http://bstavroulakis.com)

# Overview



Basic Shell

Setup our environment

Create a simple drawer system

Basics of Polymer.js

# Web Applications Are Haaard

Where to Start?

Theming/  
Templating/ Colors/  
Skins

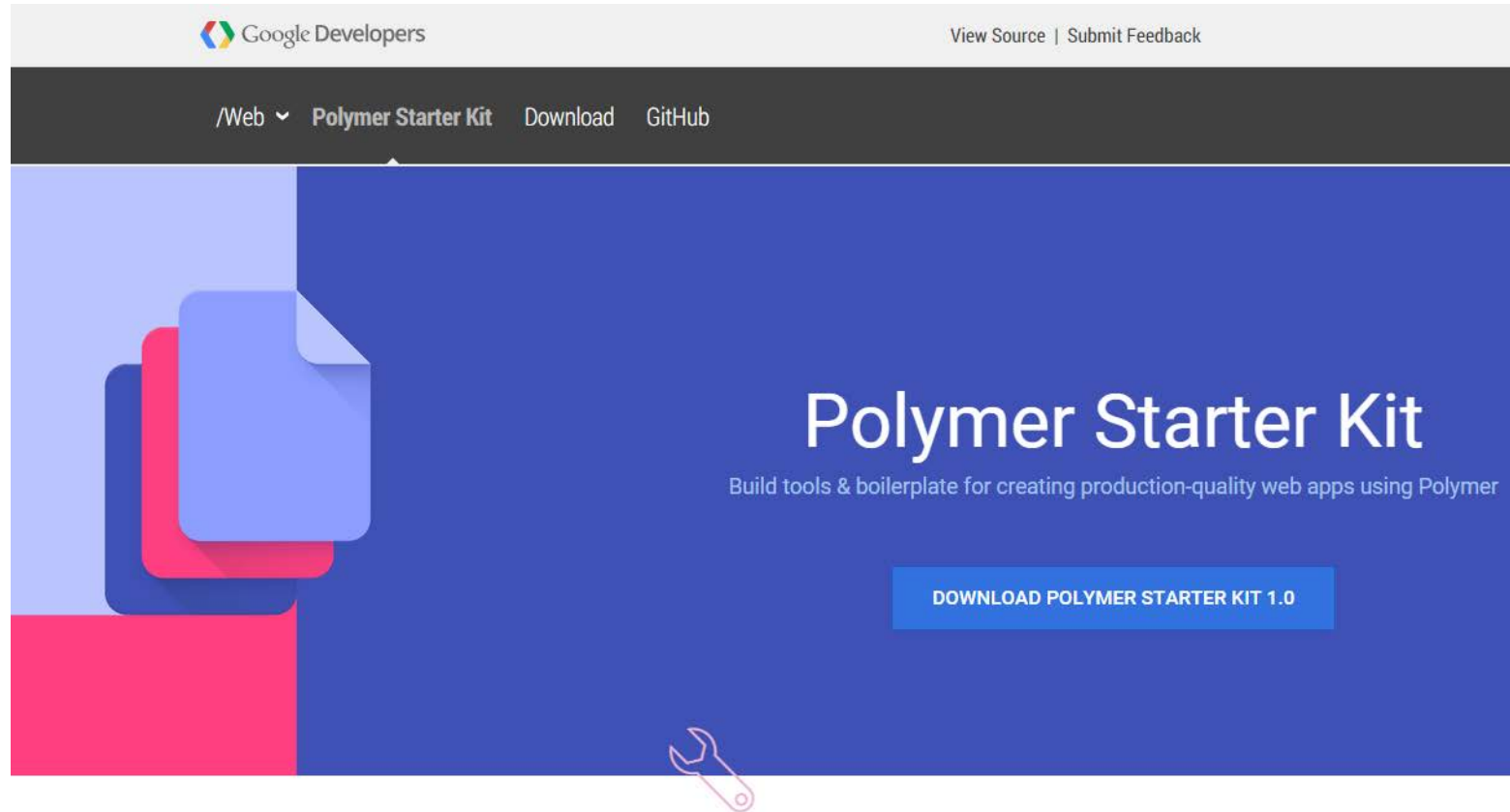
Structure

Expandable/  
Modular

Standards/  
Compatibility/  
Accessibility

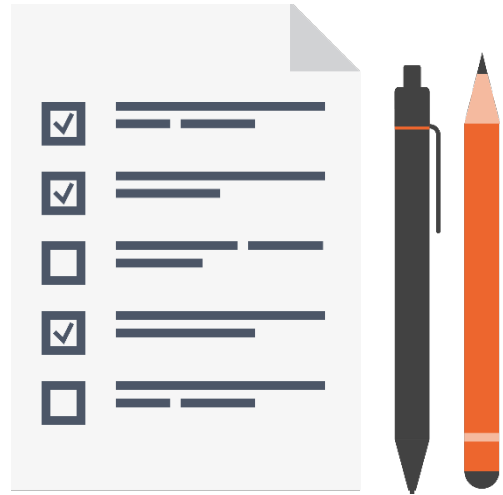
Performance and  
Much Much  
More...

# Polymer Starter Kit



<http://bit.ly/1HFBckc>

# Why Won't You Use a Starter Kit?



Setup each layer

Understanding the concepts

Decomposing layers

Polymer-Dashboard.com

A Polymer Dashboard with Material Design <http://polymer-dashboard.com/> — Edit

68 commits
1 branch
0 releases
1 contributor

branch: master
polymer-dashboard / +

added tests		
bstavroulakis authored 2 days ago latest commit a5a068730f		
app	added tests	2 days ago
demo-server	added tests	2 days ago
test	added tests	2 days ago
.gitignore	gulp-insert	10 days ago
README.md	Update README.md	6 days ago
bower.json	added tests	2 days ago
gulpfile.js	added tests	2 days ago
package.json	vulcanization	4 days ago

<http://bit.ly/1HAzR0b>

# Just My Personal Setup

## Windows 10

(Mac or Linux work perfectly as well for this project)

## Visual Studio Code

(any other HTML/CSS/JS editor is perfect)

## Node.js for Our Mockup Back-end

[\(https://nodejs.org/\)](https://nodejs.org/)

## Git Bash for Our Console

(Terminal for Linux/ Mac)

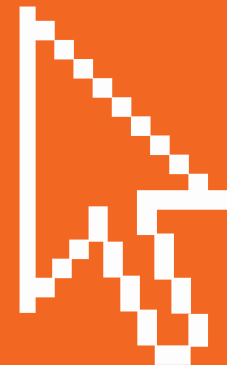
[\(https://msysgit.github.io/\)](https://msysgit.github.io/)

# Demo – Our First Element!

Declare our own elements

Include them on our page

Re-use them





# Data Binding

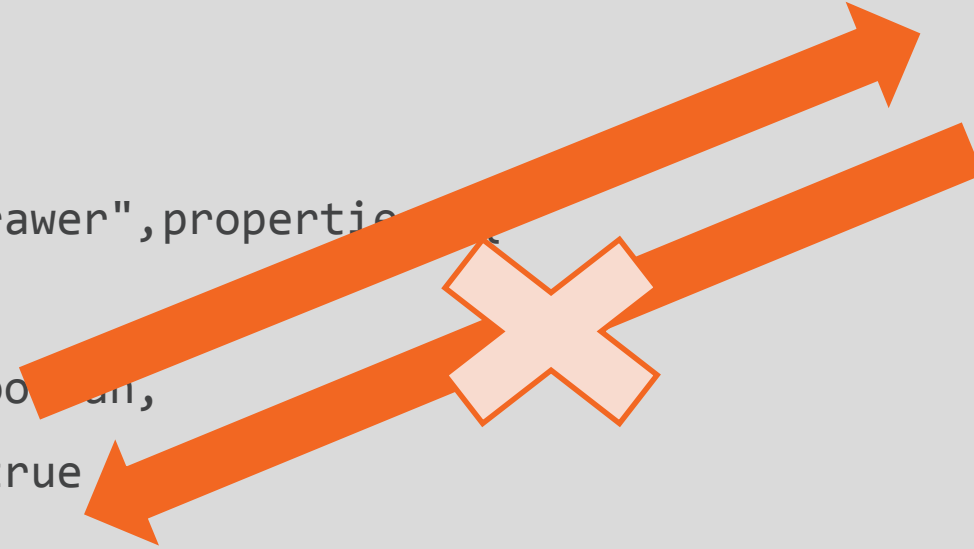
HTML



JavaScript

# Anatomy of an Element – Binding 1

```
<dom-module id="pd-drawer">
  <template>LeftShow: <input type="text" value="{{leftShow::input}}" /></template>
</dom-module>
<script>
  Polymer({is: "pd-drawer",properties: {
    leftShow: {
      type: Boolean,
      value: true
    }
  }});
</script>
```



The diagram illustrates the binding process. Two orange arrows originate from the `leftShow` property in the Polymer configuration object. One arrow points to the `LeftShow:` text in the template, and the other points to the `value="{{leftShow::input}}"` attribute of the `<input>` element. A large orange 'X' is placed over the Polymer configuration object, indicating that the configuration is not directly used for binding; instead, the property is mapped to the template and the input value.

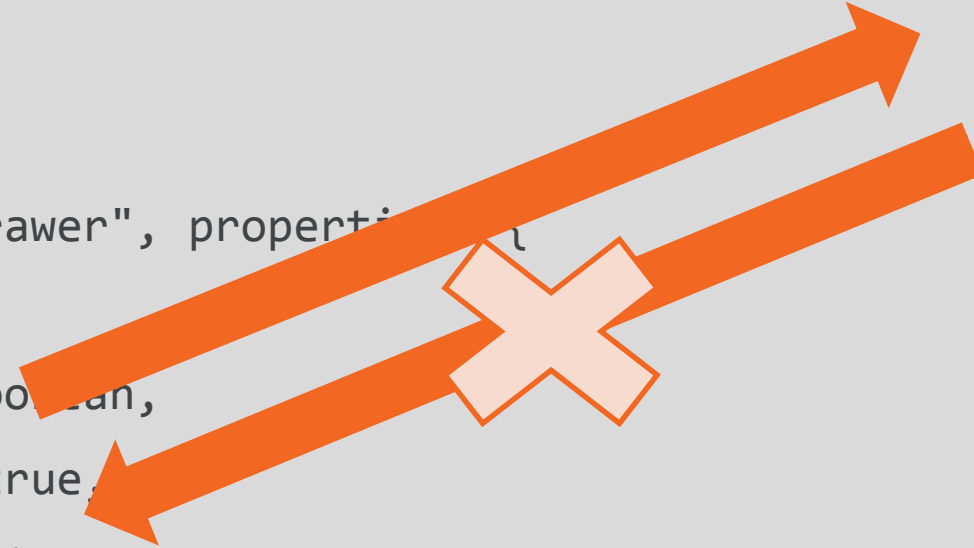
# Anatomy of an Element – Binding 2

```
<dom-module id="pd-drawer">
  <template>LeftShow: <input type="text" value="{{leftShow::input}}" /></template>
</dom-module>
<script>
  Polymer({is: "pd-drawer", properties: {
    leftShow: {
      type:Boolean,
      value:true,
      notify:true
    }
  }}});
</script>
```



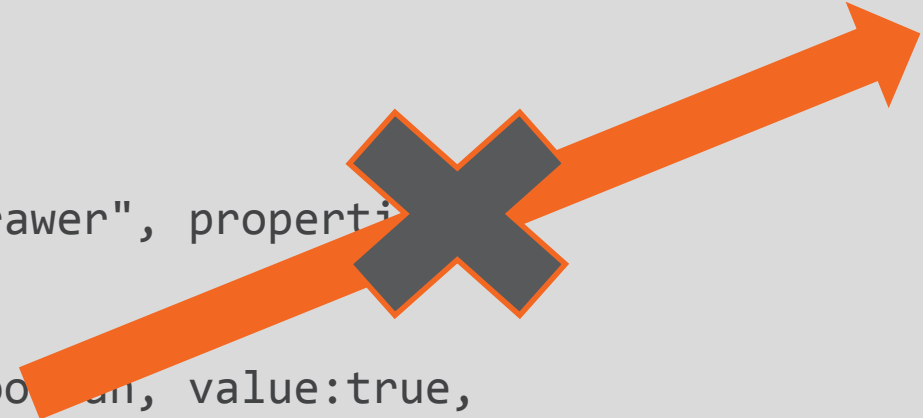
# Anatomy of an Element – Binding 3

```
<dom-module id="pd-drawer">
  <template>LeftShow: <input type="text" value="[[leftShow]]" /></template>
</dom-module>
<script>
  Polymer({is: "pd-drawer", properties: {
    leftShow: {
      type: Boolean,
      value: true,
      notify: true
    }
  }}));
</script>
```



# Anatomy of an Element – Binding 4

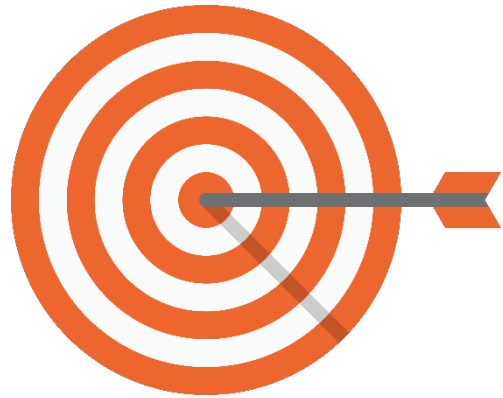
```
<dom-module id="pd-drawer">
  <template>LeftShow: <input type="text" value="{{leftShow::input}}" /></template>
</dom-module>
<script>
  Polymer({is: "pd-drawer", properties: {
    leftShow: {
      type: Boolean, value: true,
      notify: true,
      readOnly: true
    }
  }}});
</script>
```



# Insertion Points Demo



# Summary



Main Structure

Basic Concepts of Polymer.js

Theming, Coloring, Material Design