**Webcomponents&Polymer**

**一．现状：**

打开任何一个网页查看源代码，看到的都是我们熟知的那几个html元素。例如：input、span、div、table等。在web前端我们一直遵循的是w3c制定的标准—>浏览器支持标准-->开发者根据标准支持的元素开发页面。现在有许多的前端UI框架，我们后台管理系统使用的EasyUI以及现在很流行的bootstrap等等。无论是哪种框架，最终呈现在html页面源码中地还是传统的那几个标签。

**二．未来：**

Webcomponents现已经被w3c列为草案，相信不久的将来会成为正式的浏览器标准。Webcomponents使浏览器像处理原生元素一样处理我们自定义的元素标签。这个标准的理念有类于.net中的ascx用户自定义控件。区别在于webcomponent用在前端。

**三．Webcomponents**

Webcomponents

Shadow

DOM

Html Import

TEMPLATE

模

板

自

定

义

元

素

**四．Polymer**

在2013年的Google I/O大会上，Google发布了Polymer，它是一个使用Web组件构建Web应用的类库，同时也使用了为Web构建可重用组件的新的HTML 5标准。Polymer为大部分Web组件技术提供了polyfills功能，它能让开发者在所有的浏览器支持新特性前创建自己的可重用组件。此外，Polymer提供了一系列的部件的例子，其中包括天气、时钟、股票行情和线型图。

名词解释：

Polyfills: 我理解的就是用来使不同浏览器都能兼容webcomponent标准的库。在官网<http://webcomponents.org/可以下载>webcomponents.js。不支持webcomponent的浏览器需引入webcomponents.js。

使用polymer元素:

<!-- 导入 element -->

<link rel="import" href="google-map.html">

<!-- 使用 element -->

<google-map lat="37.790" long="-122.390"></google-map>

自定义元素：

<polymer-element name="my-counter" attributes="counter">

<template>

<style> /\*...\*/ </style>

<div id="label"><content></content></div>

Value: <span id="counterVal">{{counter}}</span><br>

<button on-tap="{{increment}}">Increment</button>

</template>

<script>

Polymer({

counter: 0, // Default value

counterChanged: function() {

this.$.counterVal.classList.add('highlight');

},

increment: function() {

this.counter++;

}

});

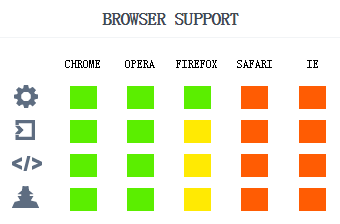
</script>

</polymer-element>

<!-- 使用 element -->

<my-counter counter="10">Points</my-counter>

**五．浏览器支持**



第一列从上到下分别代表templates、html Imports、Customer Element、ShadowDom

**六．相关网站**

WebComponents官网：http://webcomponents.org/

Polymer官网：<http://polymerchina.org/>（包含很多示例，详细介绍）

WebComponents专题：http://mobile.51cto.com/web-440551.htm