

# Modern Web Apps with HTML5 Web Components and Polymer, Java EE MVC and JAX-RS

Kito Mann (@kito99), Virtua, Inc.

# Kito D. Mann (@kito99)

- Principal Consultant at Virtua (<http://www.virtua.com>)
- Training, consulting, architecture, mentoring
  - PrimeFaces, JSF, Java EE, Web Components, Polymer, Liferay, etc.
- Official US PrimeFaces partner
- Author, JavaServer Faces in Action
- Founder, JSF Central (<http://www.jsfcentral.com>)

# Kito D. Mann (@kito99)

- Co-host, Enterprise Java Newscast (<http://enterprisejavaneews.com>)
- New site: <http://knowesis.io>
- Internationally recognized speaker
  - JavaOne, JavaZone, Devvxx, Devnexus, NFJS, etc.
- JCP Member
  - JSF, MVC, JSF Portlet Bridge, Portlets

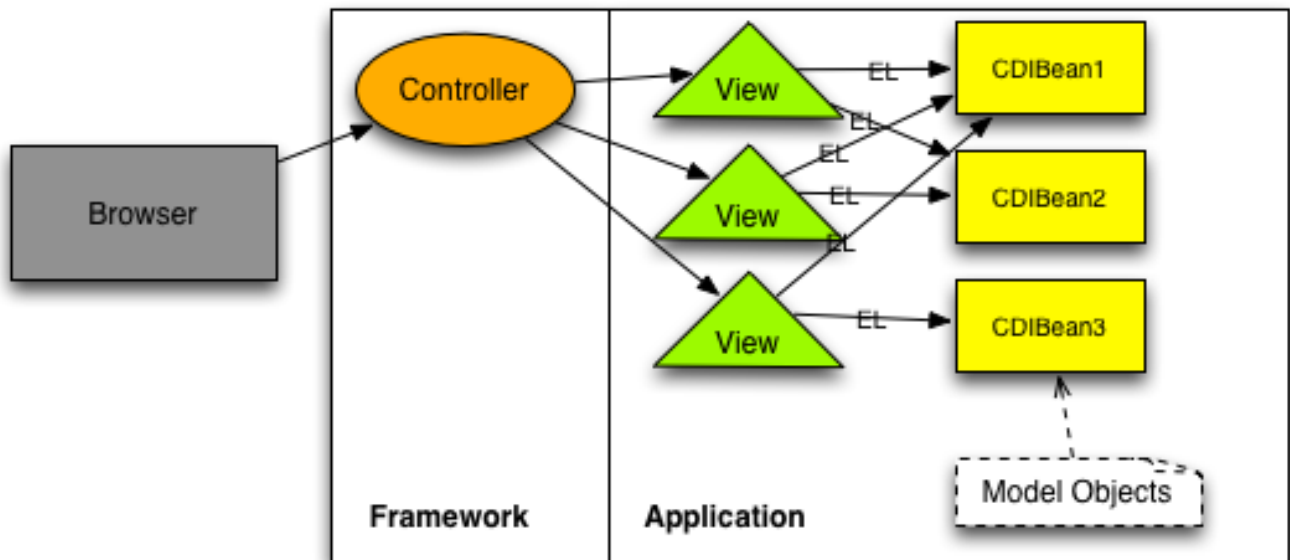
## MVC

- New [JSR \(371\)](#) for Java EE 8
- Action-based server-side framework like Spring MVC
- Based on community feedback
- Current release: EDR2
- Java EE 8 scheduled for first half of 2017

## MVC and JSF?

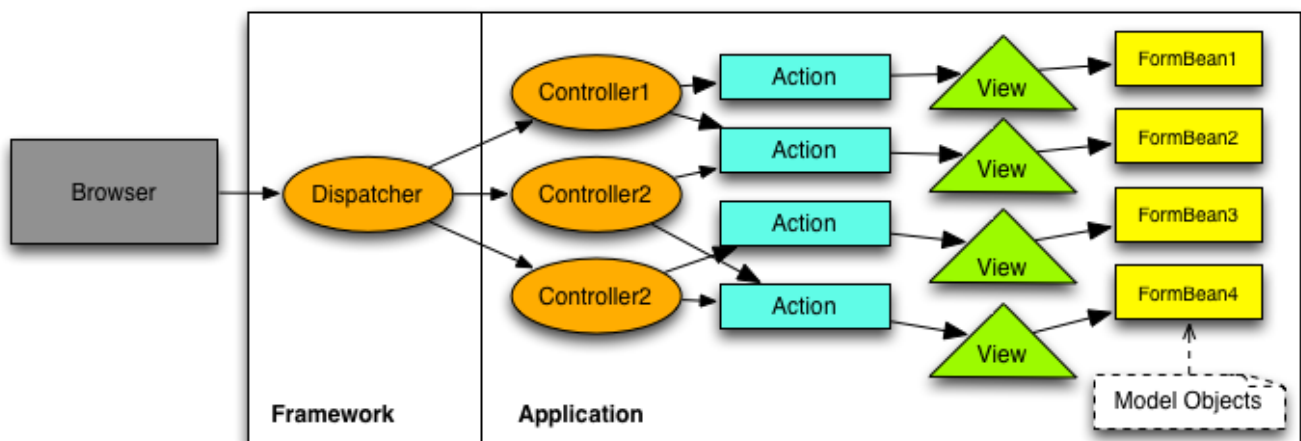
- MVC will sit alongside with JSF in Java EE (*not* a replacement)

## MVC and JSF?



Component-based frameworks (image from Ed Burns: [Why Another MVC?](#))

## MVC and JSF?



Action-based frameworks (image from Ed Burns: [Why Another MVC?](#))

## MVC

- Built on top of JAX-RS
- Controllers must use CDI
- Controllers can be in request scope
- Allows you to handle errors inside of Controllers instead of globally like JAX-RS

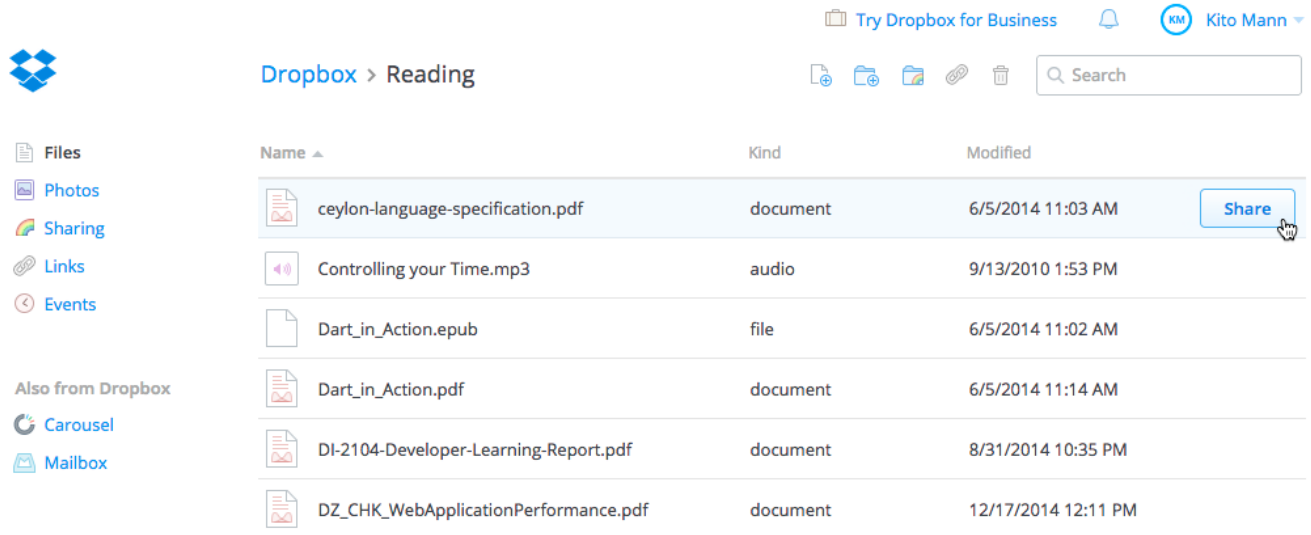
## MVC

- Bring your own template engine!
- Built-in support for CSRF protection

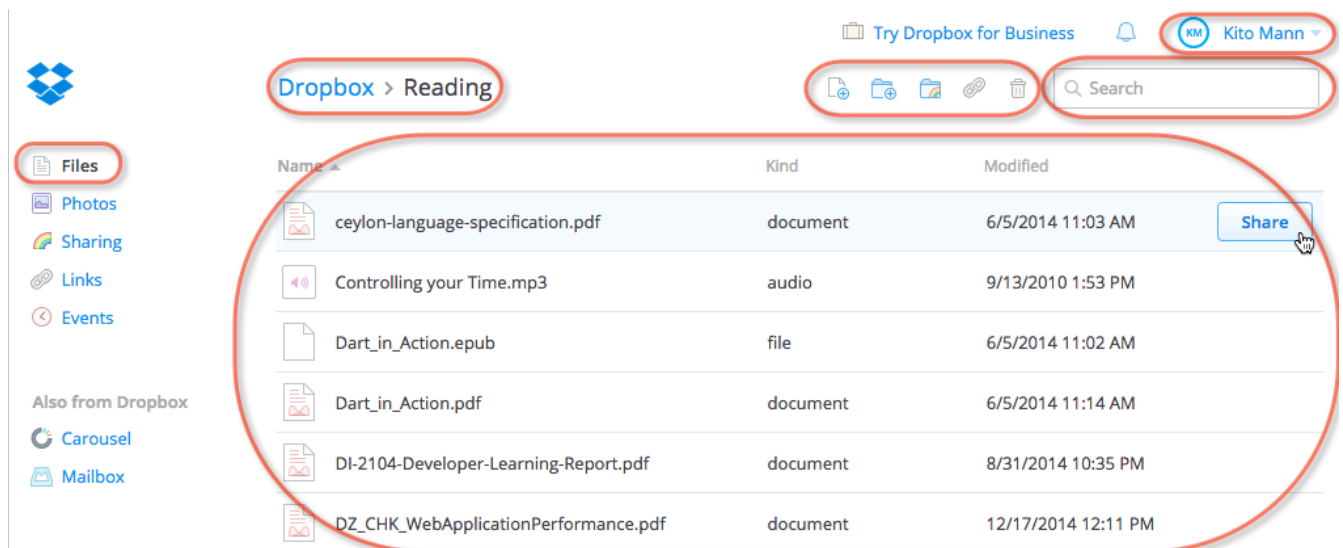
- Support for encoding to avoid XSS attacks

# What about UI Components?

## UI Components are Everywhere



## UI Components are Everywhere



## UI Components are Everywhere

- Component models have been popular since the early nineties
  - Visual Basic
  - Delphi
  - PowerBuilder
  - WinForms

- Windows Presentation Framework
- ASP.NET
- Swing
- JavaFX
- JavaServer Faces
- Tapestry

## UI Components are Everywhere

- In the browser, component suites must invent their own models:
  - YUI
  - KendoUI
  - Bootstrap
  - jQuery UI
  - Wijmo
  - PrimeUI
  - Infragistics

## Why do We Build Components?

- Reusable UI functionality
  - Within a single application
  - Across multiple applications
- You can focus on the core application functionality

## HTML Markup Doesn't Support Non-Native Components

- Programming model may be componentized, but native markup is not

## PrimeFaces (JavaServer Faces) DataTable

```

<p:dataTable var="car" value="#{dtPaginatorView.cars}" rows="10"
            paginator="true"
            paginatorTemplate="{CurrentPageReport} {FirstPageLink}
{PreviousPageLink} {PageLinks} {NextPageLink} {LastPageLink} {RowsPerPageDropdown}"
            rowsPerPageTemplate="5,10,15">
    <p:column headerText="Id">
        <h:outputText value="#{car.id}" />
    </p:column>

    <p:column headerText="Year">
        <h:outputText value="#{car.year}" />
    </p:column>

    <p:column headerText="Brand">
        <h:outputText value="#{car.brand}" />
    </p:column>

    <p:column headerText="Color">
        <h:outputText value="#{car.color}" />
    </p:column>
</p:dataTable>

```

## We Work with Abstractions

### Bootstrap Dropdowns

```

<div class="dropdown">
    <button class="btn btn-default dropdown-toggle" type="button" id="dropdownMenu1"
data-toggle="dropdown" aria-expanded="true">
        Dropdown
    <span class="caret"></span>
</button>
<ul class="dropdown-menu" role="menu" aria-labelledby="dropdownMenu1">
    <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Action</a></li>
    <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Another
action</a></li>
    <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Something else
here</a></li>
    <li role="presentation"><a role="menuitem" tabindex="-1" href="#">Separated
link</a></li>
</ul>
</div>

```

## We Work with Abstractions

### jQuery UI Tabs

```
<div id="tabs">
  <ul>
    <li><a href="#tabs-1">Nunc tincidunt</a></li>
    <li><a href="#tabs-2">Proin dolor</a></li>
    <li><a href="#tabs-3">Aenean lacinia</a></li>
  </ul>
  <div id="tabs-1">
    ...
  </div>
  <div id="tabs-2">
    ...
  </div>
  <div id="tabs-3">
    ...
  </div>
</div>
```

## What is a Web Component?

- *Web components bring a native component model to HTML*

## What is a Web Component?

### Scrolling

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

[TAP ME TO CLOSE](#)

## What is a Web Component?



```
<paper-action-dialog backdrop autoCloseDisabled layered="false">
  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor
  incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
  exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.</p>

  <paper-button affirmative autofocus>Tap me to close</paper-button>
</paper-action-dialog>
```

# Web Components == Collection of HTML5 Standards

- Custom Elements
- HTML Templates
- HTML Imports
- Shadow DOM
  - (Custom CSS Properties)

## Custom Elements

**NOTE** | Show counter and source and inspect element in DOM

[demo](#)

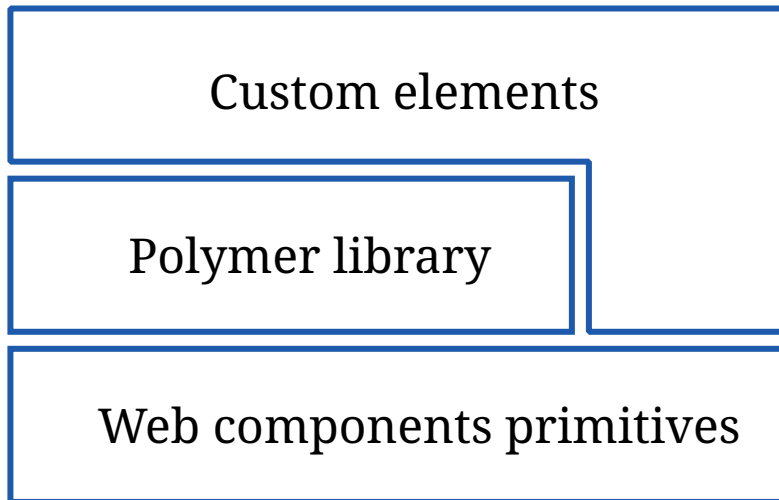
## Web Components in the Wild

- Web Components Toolkits
  - [Polymer](#)
  - [X-Tag](#)
- Component Suites
  - [Polymer Element Catalog](#)
  - [Bosonic](#)
  - [Prime Elements](#)
  - [Strand Web Components](#)

## Web Components in the Wild

- Directories
  - <http://customelements.io>

## What is Poylmer?



## Polymer

- Library for building web components
- Extensive feature set
  - Simplified programming model
  - Two-way data binding
  - Declarative event handling
  - Behaviors (mixins)
  - Property observation
- Also hosts extensive sets of web components
- Extensive set of tools
  - Build, testing, designer, etc.

## Polymer

- Developed by and used internally by Google
- Over 300 Google Projects (including <http://gaming.youtube.com>)
- Over 150,000 public facing "web pages"
- Used internally for parts of Chrome
- Heavily promoted ([Polymer Summit](#), [Polycasts](#), etc.)

# Polymer

```
<dom-module id="seed-element">

  <template>
    <style>
      :host {
        display: block;
        box-sizing: border-box;
      }
    </style>
    <content></content>
    <p class="author">
      
      Cheers, {{author.name}}!
    </p>
  </template>

</dom-module>

<script>

  Polymer({

    is: 'seed-element',

    properties: {
      fancy: Boolean,

      author: {
        type: Object,
        value: function() {
          return {
            name: 'Dimitri Glazkov',
            image: 'http://addyosmani.com/blog/wp-
content/uploads/2013/04/unicorn.jpg',
          };
        }
      },
    },

    // Element Lifecycle

    ready: function() {
      // 'ready' is called after all elements have been configured, but
      // propagates bottom-up. This element's children are ready, but parents
      // are not.
    },
  });
}
```

```

attached: function() {
  // `attached` fires once the element and its parents have been inserted
  // into a document.
},

detached: function() {
  // The analog to `attached`, `detached` fires when the element has been
  // removed from a document.
},

// Element Behavior

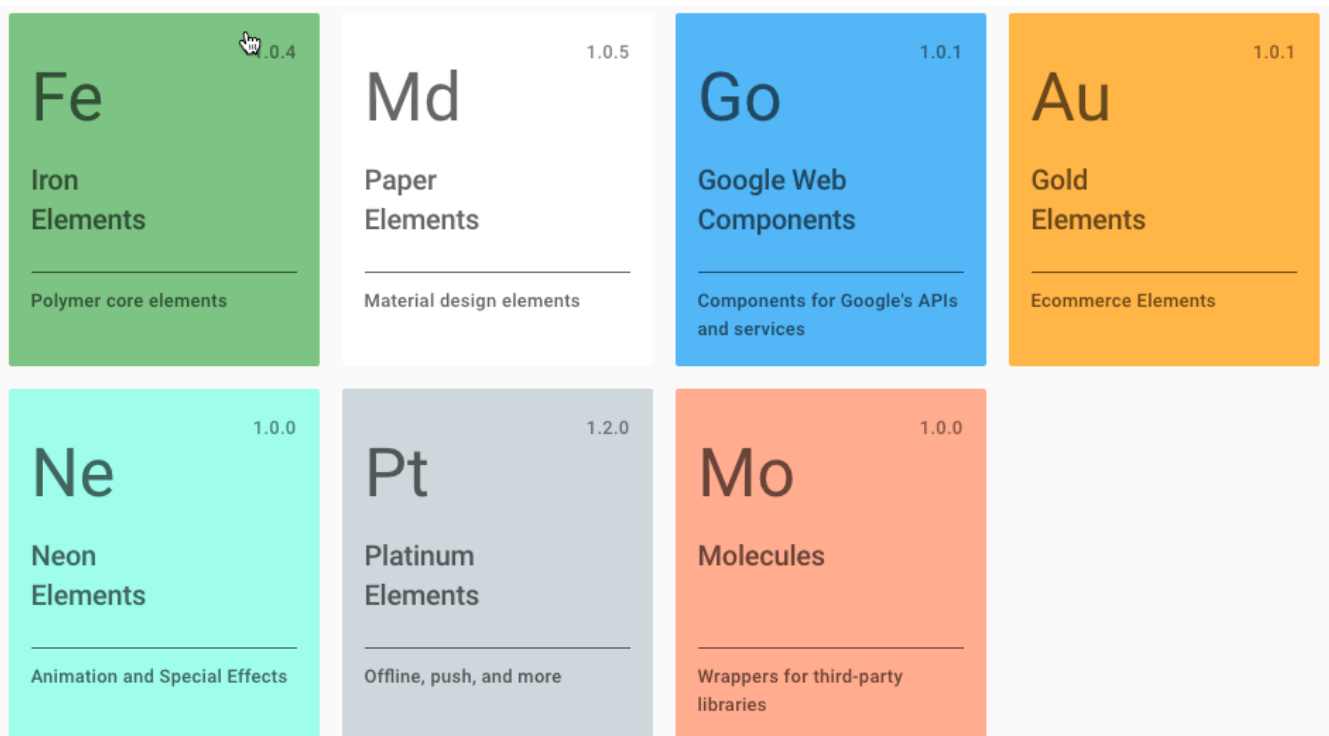
sayHello: function(greeting) {
  var response = greeting || 'Hello World!';
  return 'seed-element says, ' + response;
}

});

</script>

```

## Polymer Element Catalog



## Polymer Element Catalog

[demo](#)

**NOTE** | Show polymer starter kit too

## Sample Application: Polymer with Java EE MVC

demo ([github](#))

**NOTE** | Mention type conversion