Dev10

Unleash the Power of REST APIs with Dojo

Richard Moy

-
- Managing Director, Phora Group
- Microwave Engineer for 10 years
- Business Process Analyst for past 20 years
- IBM Champion 2015, 2016, 2017
- Working with JavaScript for past 7 years
- Chief Architect of iPhora Touch

Phora Group

- iPhora Touch

  Simple, Secure, Purpose Driven Social Collaboration

- iPhora Foundation

  Secure Automation of Mission Critical Business Processes

Agenda

- Dojo Toolkit Framework?
- Dojo/Dijit /Classes/Inheritance
- Business Requirement
- Enhance Dojo with Other Open Source Projects
- Use Case for Dojo
- Understanding Dojo Lifecycle
- Data Binding and React
- Data Modeling and Domino REST API
- Summary

Dojo Toolkit Framework

- 2004, Granddaddy of JS Frameworks
- Dojo v1.9.2 included with XPages/Domino 9.01
- Released in 2005
- Open Source Project
- Allows for custom builds
- Designed for enterprise
- Latest version Dojo 1.12
- Dojo 2 Summer of 2017

Dojo Toolkit Framework

- Extensive JS function library
- Deferred/promises
- I/O
- Extensive API
- Dijit (controller/widgets)
- AMD, (Dojo 1.7+)
- Multi-datastore support
- React/Observables/Events/Subscribe
- Classes and superclasses
- TypeScript support

Dojo Toolkit Framework

Classes

Named Classes:

```
declare("MyClass”, null, {
    // Custom properties and methods here
});
```

Anonymous Classes:

```
var MyClass = declare(null, {
    // Custom properties and methods here
})
```

Dojo Toolkit Framework

Superclasses

```
var MyClass = declare(null, {
    // Custom properties and methods here
})

var MySubClass = declare(MyClass, {
    // MySubClass now has all of MyClass's properties and methods
    // These properties and methods override parent's
});
```

Dojo Toolkit Framework

Classes and Inheritance

```
define([
   "dojo/_base/declare",
   "dijit/form/Button"
], function(declare, Button){
   return declare("mywidgetsButton", Button, {
      label: "My Button",
      onClick: function(evt){
         console.log("I was clicked!");
         this.inherited(arguments);
      }
   });
});
```

Business Requirement

Web Page Generation:

- Server-based generation of web page
  - PHP
  - XPages/JSF
  - ASP
  - Node/Express
- Server/REST/JavaScript /MVC
  - Angular
  - React
  - Vue

Business Requirement - Goals

- Separate the form from the data
- Client and mobile devices
- Provide the same control of the UI in par with server-
- based generated UI
- Create a XML-based tool to create a JavaScript-based
- application
- XML-based directives compiled to JavaScript

Business Requirement - Goals

- Create a secure MVC framework
- Scalable
- Ability to fully control DOM creation and manipulation
  - True Single Page Application
  - Dynamic Routing

Business Requirement - Goals

- Dynamic destruction of DOM and scope based on ACL and permissions
- Dynamic creation of DOM and scope based on user ACL and permissions at any given moment

Business Requirement - Goals

- Responsive controller/widgets
- Ability to fully control look, feel and functionality
- Multi-state controller/widgets
- Lightweight controller/widgets

Business Requirement

UX Pages and the DLEX Project, July 2008

```
<ux:ipField id=“field1” required=“true” value=“hello”/>
```

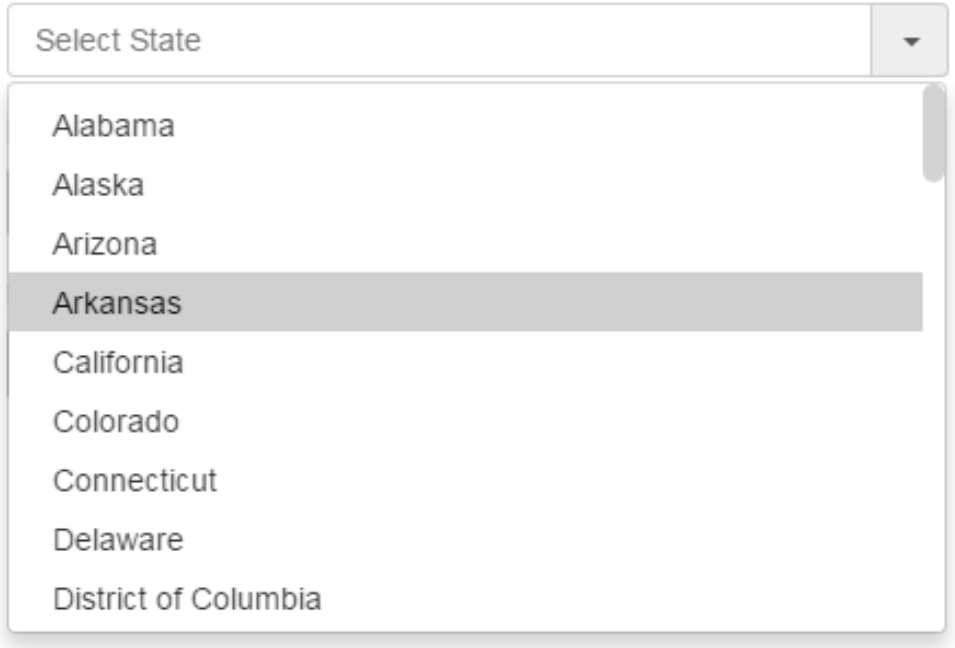
XPages, Server-based XML language for Domino

```
<xp:inputText id="field1" defaultValue="hello“ required="true">
</xp:inputText>
```

Business Requirement

Dojo Dijit-based Widget

Dojo-based Bootstrap Widget

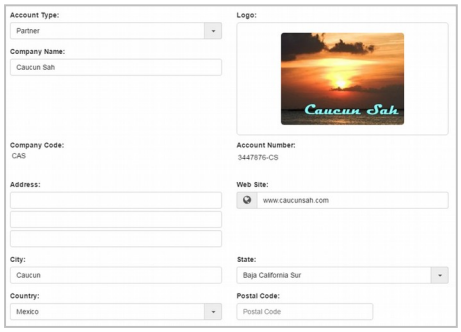


Enhance Dojo with Open Source Projects

- Blueimp, *Sebastian Tschan*
- Bootstrap, *Twitter*
- Chartjs, *Simon Brunel*
- Fontawesome, *Dave Gandy*
- Wysihtml5, *Xing*
- VMasker, *Fernando Fleury*

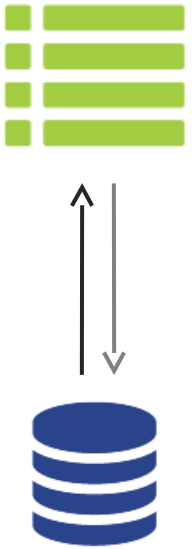
Use Case for Dojo

JavaScript/REST



JSON REST SERVICES

GET, PUT, POST, DELETE



Use Case for Dojo

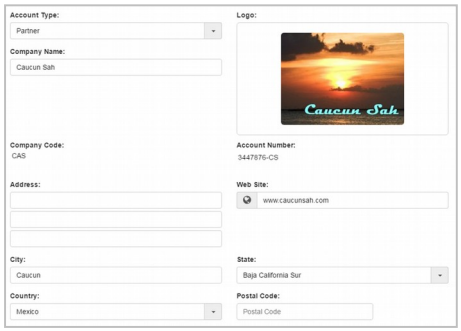
REST Services

vs

RESTful Request Services

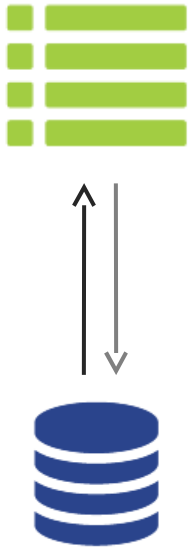
Use Case for Dojo

JavaScript/REST



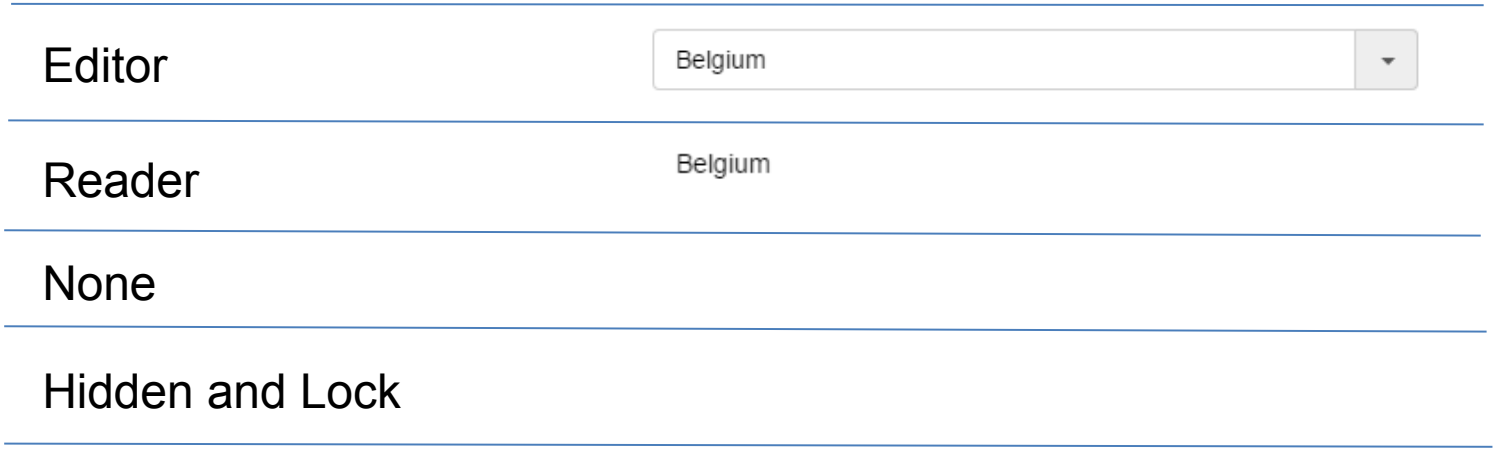
JSON REST SERVICES

GET, POST



Use Case for Dojo

- Multi-state controller/widgets



Use Case for Dojo

Account Type:
Partner

Company Name:
Caucun Sah

Logo:

Company Code:
CAS

Account Number:
3447876-CS

Address:

Web Site:
www.caucunsah.com

City:
Caucun

State:
Baja California Sur

Country:
Mexico

Postal Code:
Postal Code

Use Case for Dojo

Account Type:

Partner

Company Name:

Caucun Sah

Logo:

Company Code:

Account Number:

Address:

Web Site:

www.caucunsah.com

City:

Caucun

State:

Baja California Sur

Country:

Mexico

Postal Code:

Postal Code

Use Case for Dojo

Account Type:

Partner

Company Name:

Caucun Sah

Logo:

Company Code:

Account Number:

Address:

Web Site:
www.caucunsah.com

City:
Caucun

State:
Baja California Sur

Country:
Mexico

Postal Code:

Creating a MVC

- app ---
  - *scope*
  - views ---
    - *dialog box*
    - *view panes*
      - containers ---
        - *panes*
        - *forms*
          - controllers/widgets
            - *calendar*
            - *combobox*
            - *grid*
            - *list*
            - *textbox*
            - *uploader*

Dojo Dijit

- Core framework for creating Dojo controllers / widgets
- Defines controller lifecycle
- Defines controller scaffolding
- Declarative/Programmatically

Understanding Dojo Lifecycle

- constructor
- parameters are mixed into the widget instance
- postMixinProperties
- buildRendering
- process initial setters/getters
- postCreate
- startup
- destroy

Understanding Dojo Lifecycle

- Custom lifecycles
- Custom setters/getters

More Demos

Demos and Code

Dojo Data Binding and Reaction

- Observables
- On/emit
- Topic/subscribe
- Watch

More Demos

Demos and Code

Data Modeling with Dojo Store/ Domino DAS

- Dojo Store/Memory
- Custom Scope/Binding

More Demos

Super Demo with Domino DAS REST

Dojo / Bootstrap / Observables / React / Data Binding / Subscribe / Fontawesome / Chartjs

More Demos

Demo with Blueimp

Dojo / Bootstrap / Observables / React / Data Binding / Subscribe / Fontawesome / Blueimp

Summary

Pros of Dojo

- Extremely power and flexible
- True JavaScript framework
- Allows developers tailor to meet their needs
- Scalable
- Designed for enterprise
- Enforces discipline and design patterns

Summary

- Cons of Dojo
  - Documentation is not as robust
  - Not for novices
  - Significant learn curve compared to other frameworks
  - Best that one understands object oriented programming

Contact Information

- Twitter - @richardmoy
- Blog - http://dominointerface.com
- Web site – http://www.phoragroup.com