Contents

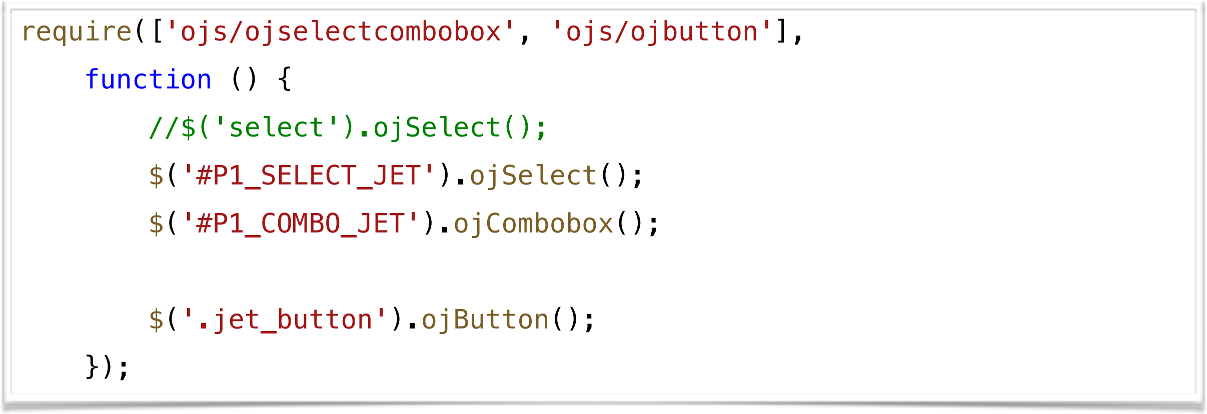
Initialiser

In Page Designer

- Add JavaScript File URL to page:
  `[require jet]/i/adbc/apex-jet-integration/initialiser/main.js`
- Add CSS File URL to page:
  `/i/libraries/oraclejet/2.0.2/css/libs/oj/v2.0.2/alta/oj-alta-min.css`
- Create Select List Items and Buttons.

On WebServer

- Create file:
  `/i/adbc/apex-jet-integration/initialiser/main.js`
  Use the item names as the jQuery selector to initialise the objects



```
require(['ojs/ojselectcombobox', 'ojs/ojbutton'],
    function () {
        //$('select').ojSelect();
        $('#P1_SELECT_JET').ojSelect();
        $('#P1_COMBO_JET').ojCombobox();

        $('.jet_button').ojButton();
    });
```

Pros

- Very Fast
- Lots of new features
- Minimal effort in APEX

Cons

- Side affects of JET CSS
- Does only work when APEX Item has the same base structure as the JET Element

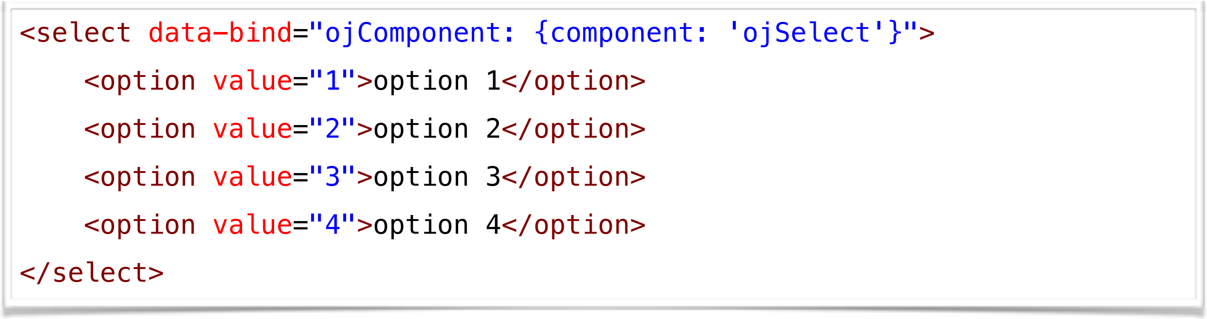
External Documentation

- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojSelect.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojCombobox.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojButton.html

Knockout

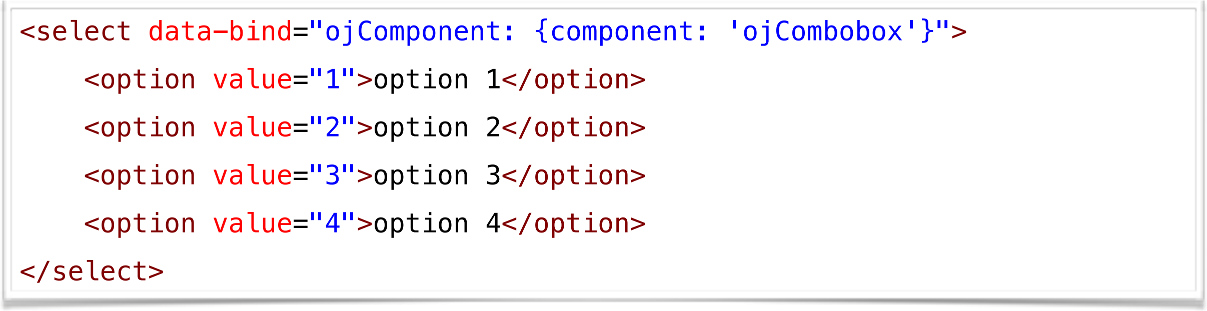
In Page Designer

- Add JavaScript File URL to page:
  `[require jet]/i/adbc/apex-jet-integration/knockout-items/main.js`
- Add CSS File URL to page:
  `/i/libraries/oraclejet/2.0.2/css/libs/oj/v2.0.2/alta/oj-alta-min.css`
- Create Static Content Region
  Content for ojSelect:



```
<select data-bind="ojComponent: {component: 'ojSelect'}">
    <option value="1">option 1</option>
    <option value="2">option 2</option>
    <option value="3">option 3</option>
    <option value="4">option 4</option>
</select>
```

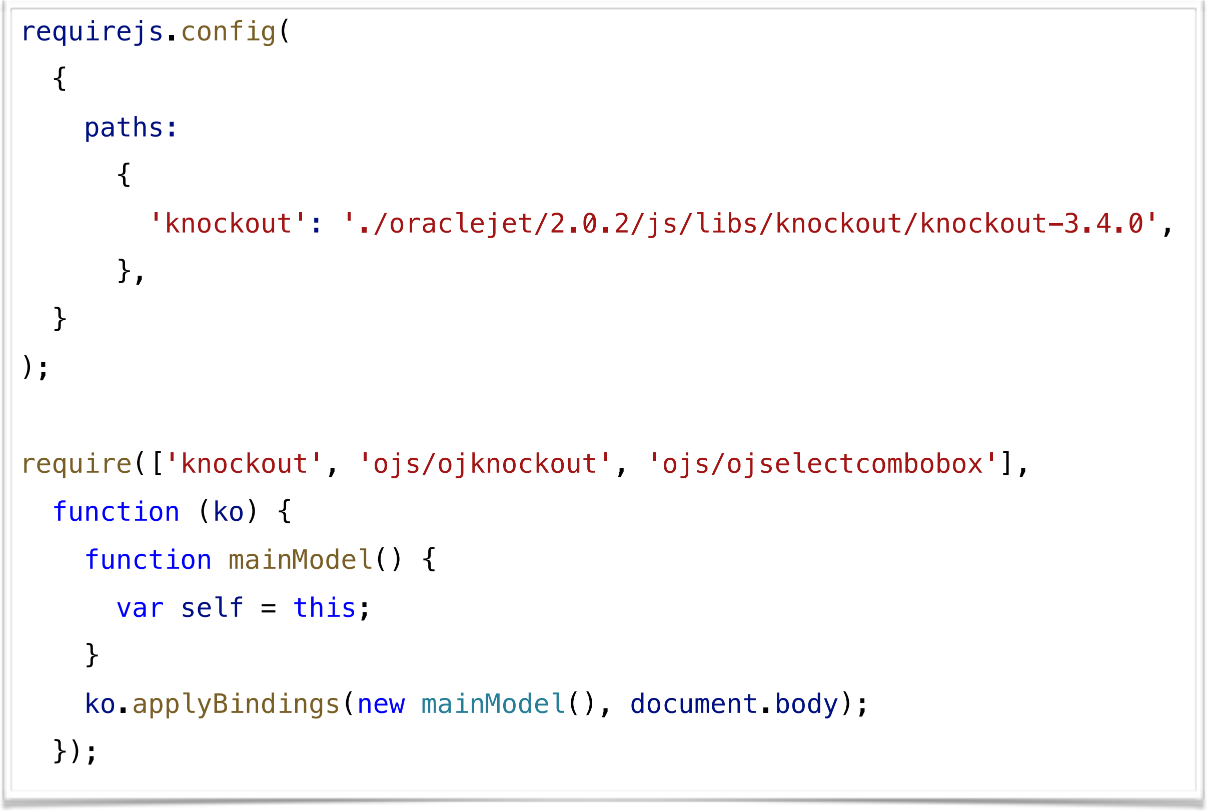
Content for ojCombobox:



```
<select data-bind="ojComponent: {component: 'ojCombobox'}">
    <option value="1">option 1</option>
    <option value="2">option 2</option>
    <option value="3">option 3</option>
    <option value="4">option 4</option>
</select>
```

On WebServer

- Create file:
  `/i/adbc/apex-jet-integration/knockout-items/main.js`
  Enable Knockout in require and bind it to the document body.



```
requirejs.config(
  {
    paths:
      {
        'knockout': './oraclejet/2.0.2/js/libs/knockout/knockout-3.4.0',
      },
  }
);

require(['knockout', 'ojs/ojknockout', 'ojs/ojselectcombobox'],
  function (ko) {
    function mainModel() {
      var self = this;
    }
    ko.applyBindings(new mainModel(), document.body);
  });
```

Pros

- Fast
- Flexible
- Lots of new features

Cons

- Side affects of JET CSS
- Every item is a region

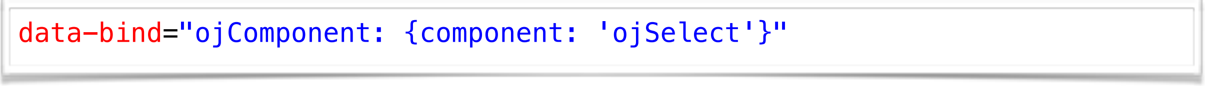
External Documentation

- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojSelect.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojCombobox.html

Data Attribute (item)

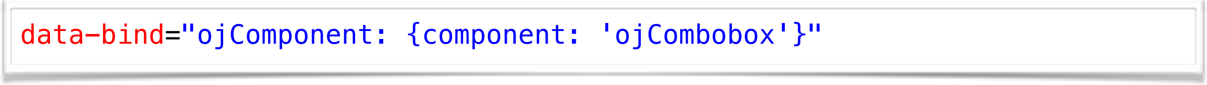
In Page Designer

- Add JavaScript File URL to page:
  `[require jet]/i/adbc/apex-jet-integration/data-attribute/main.js`
- Add CSS File URL to page:
  `/i/libraries/oraclejet/2.0.2/css/libs/oj/v2.0.2/alta/oj-alta-min.css`
- Create Select List Item
  Content for Advanced Custom Attributes for ojSelect:



```
data-bind="ojComponent: {component: 'ojSelect'}"
```

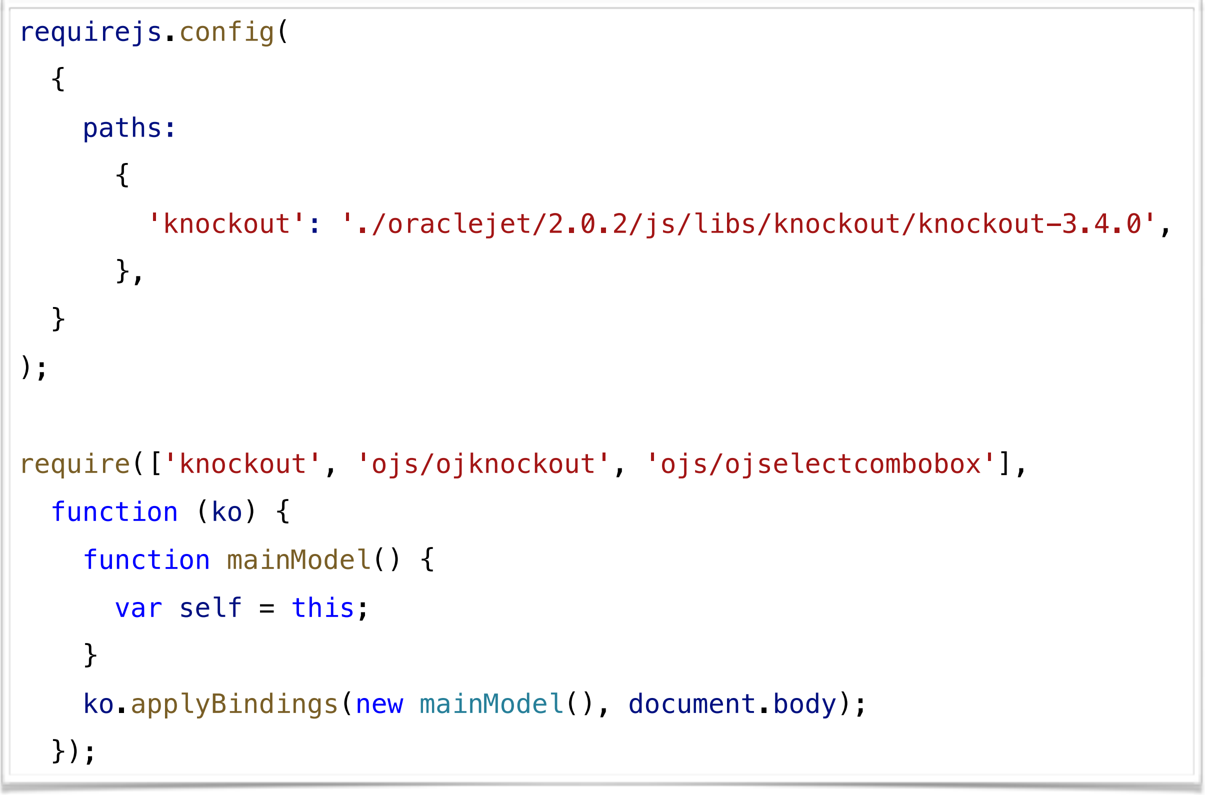
Content for Advanced Custom Attributes for ojCombobox:



```
data-bind="ojComponent: {component: 'ojCombobox'}"
```

On WebServer

- Create file:
  `/i/adbc/apex-jet-integration/data-attribute/main.js`
  Enable Knockout in require and bind it to the document body.



```
requirejs.config(
  {
    paths:
      {
        'knockout': './oraclejet/2.0.2/js/libs/knockout/knockout-3.4.0',
      },
  }
);

require(['knockout', 'ojs/ojknockout', 'ojs/ojselectcombobox'],
  function (ko) {
    function mainModel() {
      var self = this;
    }
    ko.applyBindings(new mainModel(), document.body);
  });
```

Pros

- Very Fast
- Very Flexible
- Lots of new features

Cons

- Side affects of JET CSS
- Does only work when APEX Item has the same base structure as the JET Element

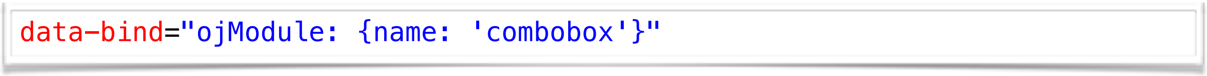
External Documentation

- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojSelect.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojCombobox.html

Data Attribute (region)

In Page Designer

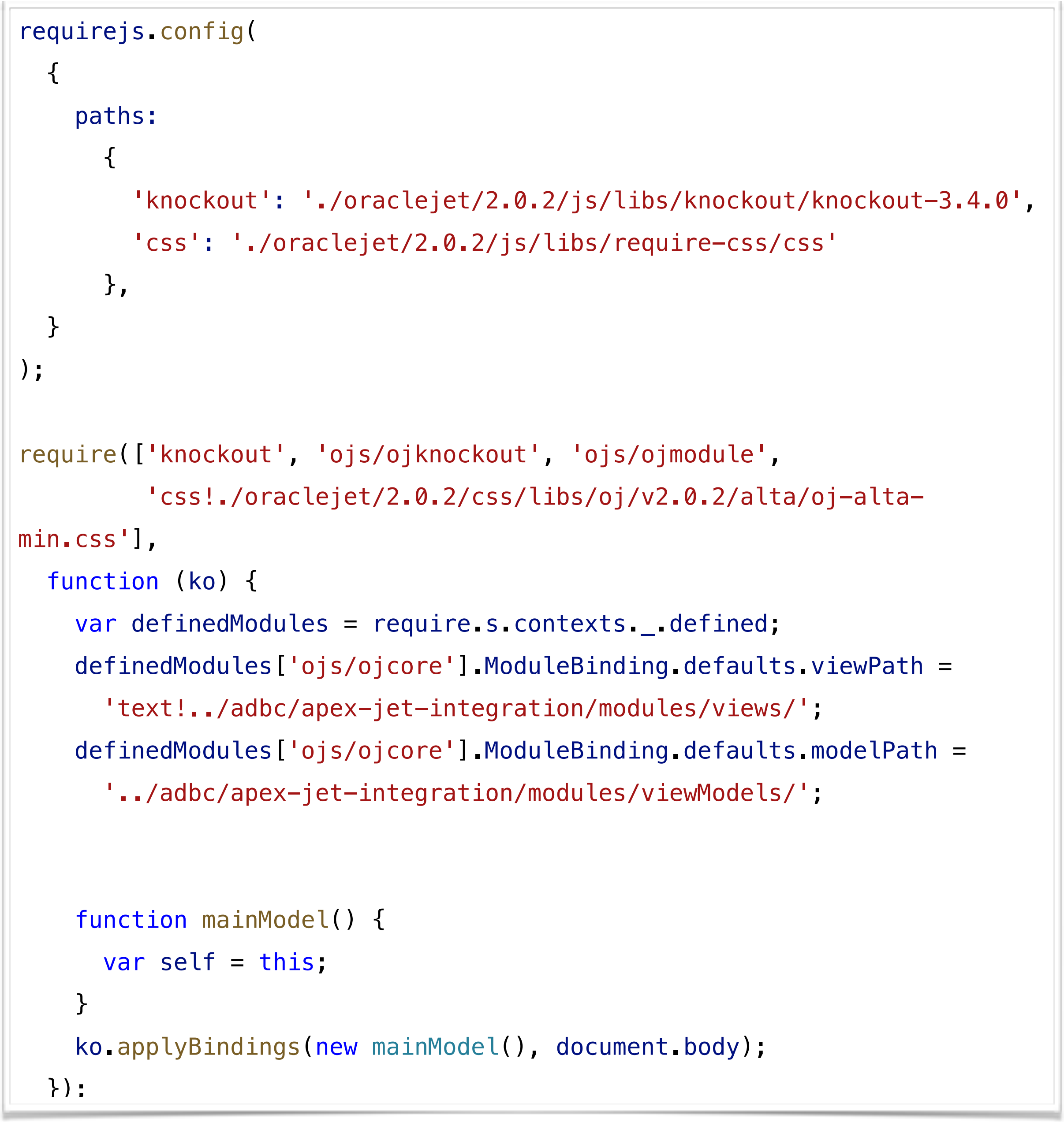
- Add JavaScript File URL to page:
  `[require jet]/i/adbc/apex-jet-integration/modules/main.js`
- Create Static Content Region
  Content for Layout Column Attributes:



```
data-bind="ojModule: {name: 'combobox'}"
```

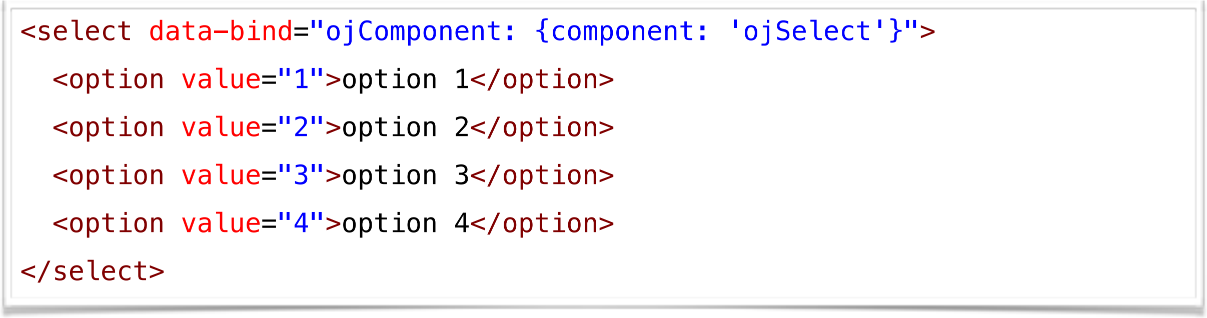
On WebServer

- Create file:
  `/i/adbc/apex-jet-integration/modules/main.js`
  Enable Knockout in require and bind it to the document body.
  Enable css file loading to add custom css to the module.
  Change the default module path (APEX doesn’t use this).



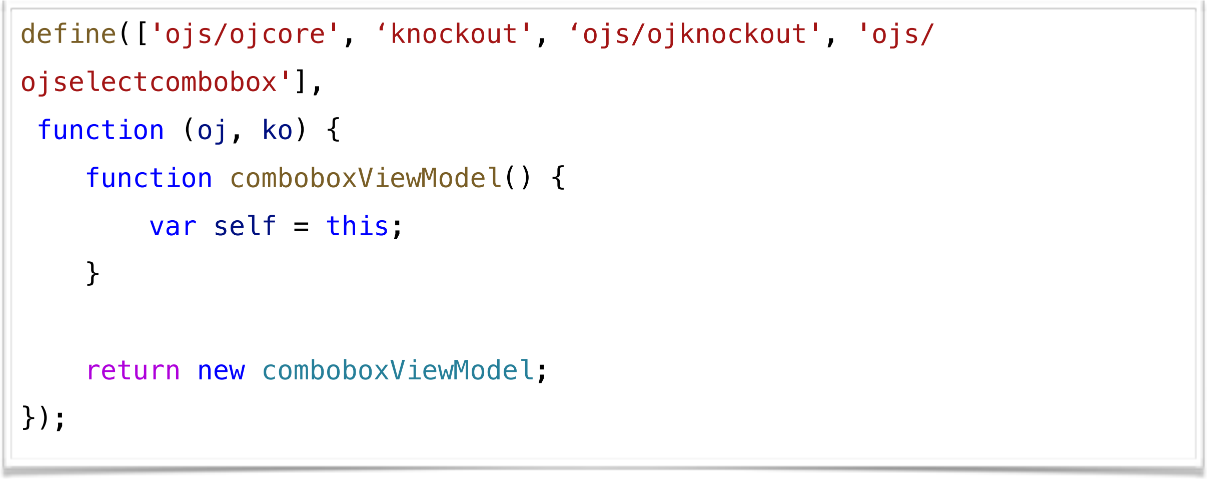
```
requirejs.config(
  {
    paths:
      {
        'knockout': './oraclejet/2.0.2/js/libs/knockout/knockout-3.4.0',
        'css': './oraclejet/2.0.2/js/libs/require-css/css'
      },
  }
);


require(['knockout', 'ojs/ojknockout', 'ojs/ojmodule',
         'css!./oraclejet/2.0.2/css/libs/oj/v2.0.2/alta/oj-alta-
min.css'],
  function (ko) {
    var definedModules = require.s.contexts._.defined;
    definedModules['ojs/ojcore'].ModuleBinding.defaults.viewPath =
      'text!../adbc/apex-jet-integration/modules/views/';
    definedModules['ojs/ojcore'].ModuleBinding.defaults.modelPath =
      '../adbc/apex-jet-integration/modules/viewModels/';



    function mainModel() {
      var self = this;
    }
    ko.applyBindings(new mainModel(), document.body);
  });
```

- Create view file:
  /i/adbc/apex-jet-integration/modules/views/



```
<select data-bind="ojComponent: {component: 'ojSelect'}">
  <option value="1">option 1</option>
  <option value="2">option 2</option>
  <option value="3">option 3</option>
  <option value="4">option 4</option>
</select>
```

- Create viewModel file:
  /i/adbc/apex-jet-integration/modules/viewModel/



```
define(['ojs/ojcore', ‘knockout', ‘ojs/ojknockout', 'ojs/
ojselectcombobox'],
 function (oj, ko) {
    function comboboxViewModel() {
        var self = this;
    }

    return new comboboxViewModel;
});
```

Pros

- Fast
- Very Flexible
- Lots of new features
- High reusability of code
- Minimal effort in APEX

Cons

- Side affects of JET CSS
- No low coding

External Documentation

- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojSelect.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ojCombobox.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.Model.html
- http://www.oracle.com/webfolder/technetwork/jet-220/jsdocs/oj.ModuleBinding.html