

Introduction

Explain:

<http://www.openmv.com/OpenAjax.js>

Use Cases

OpenAjax.provide

```
OpenAjax.provide('acme.DialogTree');
```

OpenAjax.require

1. `OpenAjax.require('dijit.Dialog');`
`OpenAjax.require('YAHOO.widget.tree');`
2. `OpenAjax.require('dijit.Dialog', 'YAHOO.widget.tree');`

OpenAjax.register

Tell registry to load resource locally

```
OpenAjax.register({'acme.DialogTree': {uri: 'file://acme/widgets/dialogtree.js'}});  
OpenAjax.register({'acme.DialogTree.css': {uri: 'file://acme/widgets/dialogtree/custom.css',  
type: 'css'}});
```

Specify to load resource remotely (i.e. from a CDN)

```
OpenAjax.register({'acme.DialogTree': {uri: 'http://cdn.acme.com/widgets/dialogtree.js'}});  
OpenAjax.register({'acme.theme.blue': {uri: 'http://cdn.acme.com/themes/blue?v6.0', type: 'css'}});
```

Example scenario (acme)

Basic loading - working demo

Directory layout

```
resources/dojo  
resources/dijit  
resources/YAHOO  
resources/acme  
acme-tree.html
```

```
OpenAjax.register({'acme.DialogTree': {uri: 'file://acme/ widgets/dialogtree.js'},  
'acme.DialogTree.css': {uri: 'file://acme/widgets/dialogtree/custom.css'}}
```

```
});  
OpenAjax.require('acme.DialogTree');  
var t = new acme.widget.DialogTree();  
t.show();
```

Download this example from here (link).

Loading dojo & yahoo from CDN by changing the OpenAjax registry (not implemented yet)

Directory layout

```
resources/acme  
acme-tree.html
```

```
OpenAjax.register({'dijit.Dialog': {uri: 'http://o.aolcdn.com/dojo/0.9.0beta/dijit/Dialog.js'},  
                  'YAHOO.widget.tree': {uri: 'http://cnd.yahoo.com/somewhere/tree.js'  
}, true);
```

Or

```
OpenAjax.register({'dijit': {uri: 'http://o.aolcdn.com/dojo/0.9.0beta/dijit', type: 'dir'},  
                  'YAHOO': {uri: 'http://cnd.yahoo.com/somewhere', type: 'dir'}  
}, true);
```

```
OpenAjax.require('acme.DialogTree');  
var t = new acme.widget.DialogTree();  
t.show();
```

Unclear at this point how this would work but gives a good idea of how the registry would affect the loading of resources.

Notice here the second argument is set to true which means we want to override any values already existing in the registry.

Some notes & thoughts

I am using dojo as a starting point since in my opinion it has done the best work with the concept of resource loading and planning for interoperability.

I think it's important that there is a clear separation between a resource identifier and how a particular resource should be loaded. It should be the registry's role to let the loader know how it should get this particular resource. However for practical reasons, if a resource is not found in the registry, it would be a good best practice to have the resource identifier point to a relative file path. For instance:

```
<script type="text/javascript" src="/resources/openajax/hub.js"></script>
```

```
// One directory up from the openAjax library
```

```
OpenAjax.register('loadPath', 'file://../'); or absolute path: OpenAjax.register('loadPath', 'file:///');
```

File paths are relative to **resources/**:

resources/dojo/widget.js

```
OpenAjax.require('dojo.widget');
```

resources/dijit/foo.js

```
OpenAjax.require('dijit.foo');
```

resources/YAHOO/widget/a.js

```
OpenAjax.require('YAHOO.widget.a');
```

resources/YAHOO/main.css (non-JavaScript resource)

```
OpenAjax.register({'YAHOO.css.main': {uri: 'YAHOO/main.css', type:'css'}});
```

```
OpenAjax.require('YAHOO.css.main');
```

Could support this:

```
// Load everything remotely?
```

```
OpenAjax.register('loadPath', 'http://www.acme.co/supercdn/location');
```

Some sample code of an OpenAjax registry implementation:

```
http://www.openmv.com/acme/base.js
```