
SharePoint 2010 Managed Client Object Model

Jason Cribbet

Summit 7 Systems Inc.

Senior Developer

<http://summit7systems.com/blogs/jasoncribbet>

@jcribbet



WHAT'S YOUR MOUNTAIN?

Agenda

- Intro
- Architecture
- Usage
- OM Objects
- Demo
- Benefits & Limitations



What is the SP Client Object Model?

- Object Oriented approach to access SP Data
- 3 APIs for interacting with SharePoint Sites
 - .Net App (.Net Framework 3.5+)
 - Silverlight App (2.0+)
 - ECMAScript (JavaScript, Jscript)
- Create and Modify many SharePoint objects
 - Add and Remove Lists
 - Add, Update, and Delete List Items
 - Change documents in document libraries
 - Create Sites
 - Manage Permissions of items
 - Add and remove Web Parts from page



2007 Architecture

Server Side

Web Parts, Custom Controls, etc.

Server OM

JavaScript/Managed Code

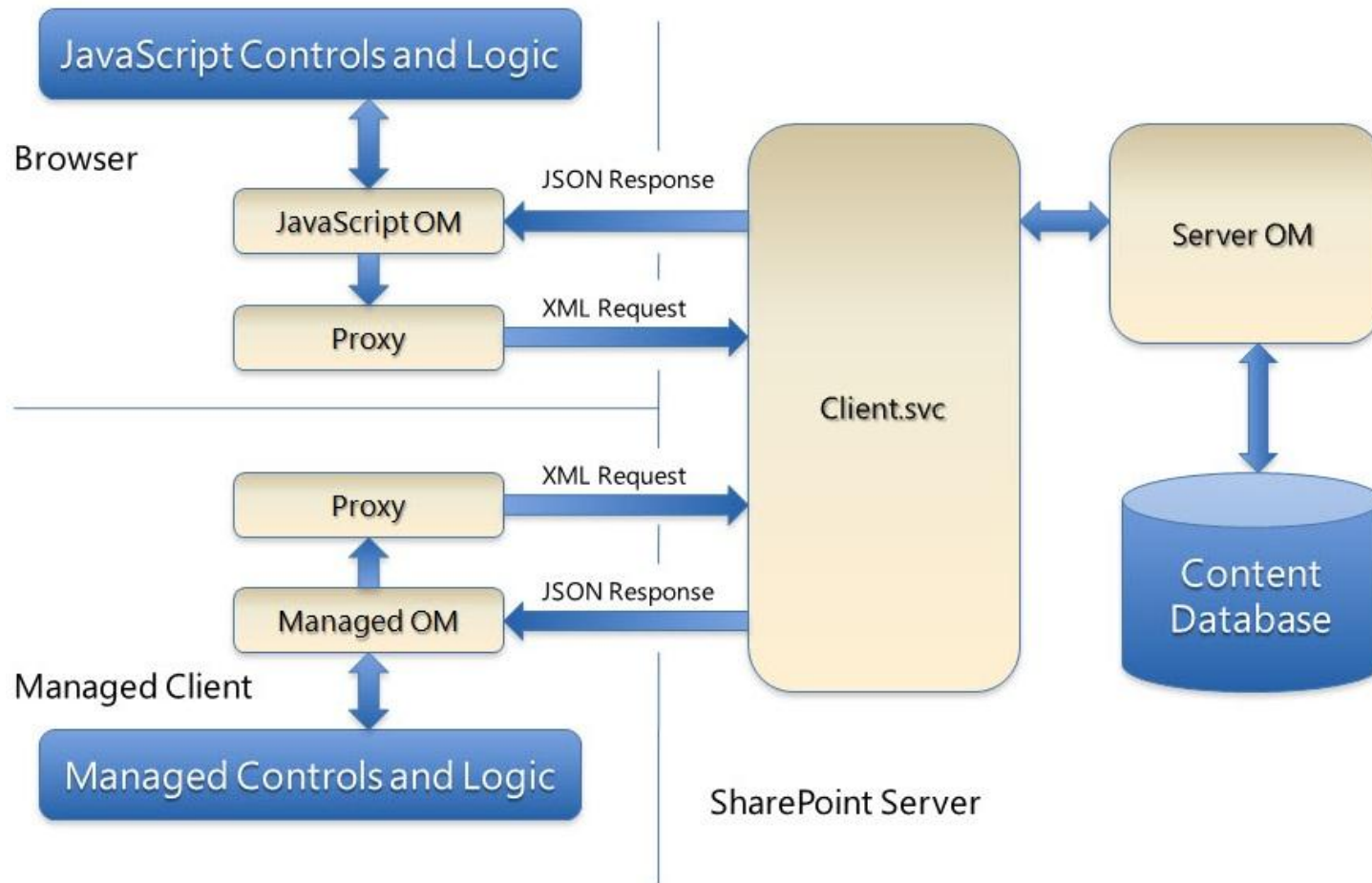
Web Services

Content Database

Client Side



2010 Architecture



Usage

- .Net Assemblies
 - Microsoft.SharePoint.Client.dll
 - Microsoft.SharePoint.Client.Runtime.dll
- ECMAScript
 - Include “sp.js” within the page

```
<Sharepoint:ScriptLink name="SP.js" runat="server" OnDemand="true" localizable="false" />
```

- Ensure “sp.js” loaded before execution

```
// Trigger the loading of data after the page completes the load of sp.js  
ExecuteOrDelayUntilScriptLoaded(populateForm, "sp.js");
```



OM Objects

- Server OM vs. Client OM

Server Side Classes	Client Side Classes
SPContext	ClientContext
SPSite	Site
SPWeb	Web
SPList	List
SPListItem	ListItem
SPField	Field



Key Code Objects

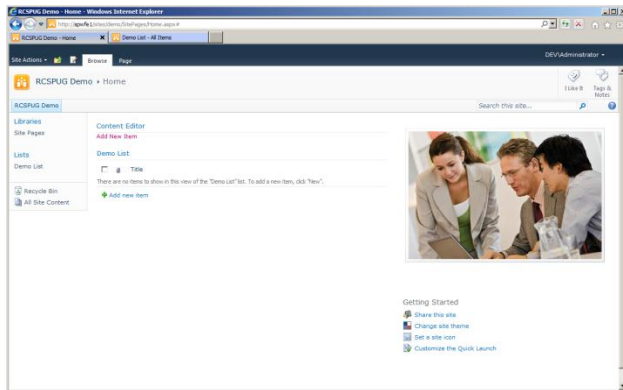
- ClientContext.Load(clientObject, retrievals);
- ClientContext.ExecuteQuery();

```
using System;
using Microsoft.SharePoint.Client;

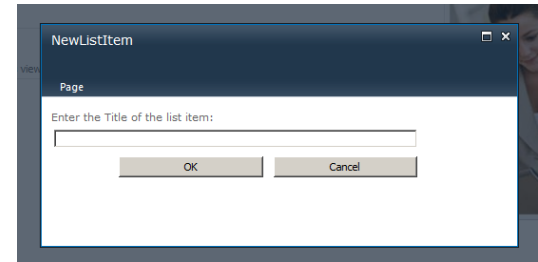
class DisplayWebTitle
{
    static void Main()
    {
        ClientContext clientContext = new ClientContext("http://intranet.contoso.com");
        Web site = clientContext.Web;
        clientContext.Load(site);
        clientContext.ExecuteQuery();
        Console.WriteLine("Title: {0}", site.Title);
    }
}
```



Demo

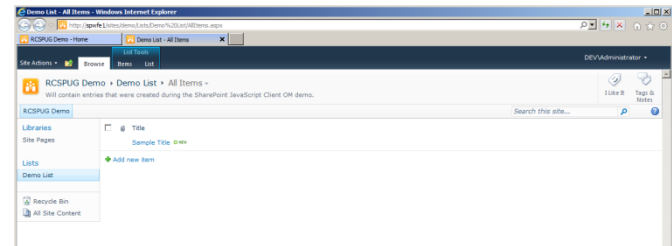


```
<script type="text/javascript">  
  //Dialog opening  
  function OpenDialog() {  
    ...  
  }  
</script>
```



```
<script type="text/javascript">  
  // Dialog callback  
  function CloseCallback(result, target) {  
    ...  
  }  
</script>
```

1. Respond to link with a SP Dialog
2. Accept user input
3. Create a List Item



Benefits

- Client OM looks similar to Server OM
- Common API across ECMA, Silverlight, and .Net managed code
- Abstracts the request details
- Can access SP data remotely
- Access data directly without using Web Services



Limitations

- No elevation of privilege
- Can't access WebApplication or Farm objects
- Requests are throttled
- Silverlight and JavaScript are asynchronous



Questions?

Jason Cribbet

Summit 7 Systems Inc.

Senior Developer

Twitter: @jcribbet

<http://summit7systems.com/blogs/jasoncribbet>



WHAT'S YOUR MOUNTAIN?