SharePoint Happenings

SHAREPOINT 2010 CLIENT SIDE OBJECT MODEL

Phil Wicklund

SharePoint Happenings

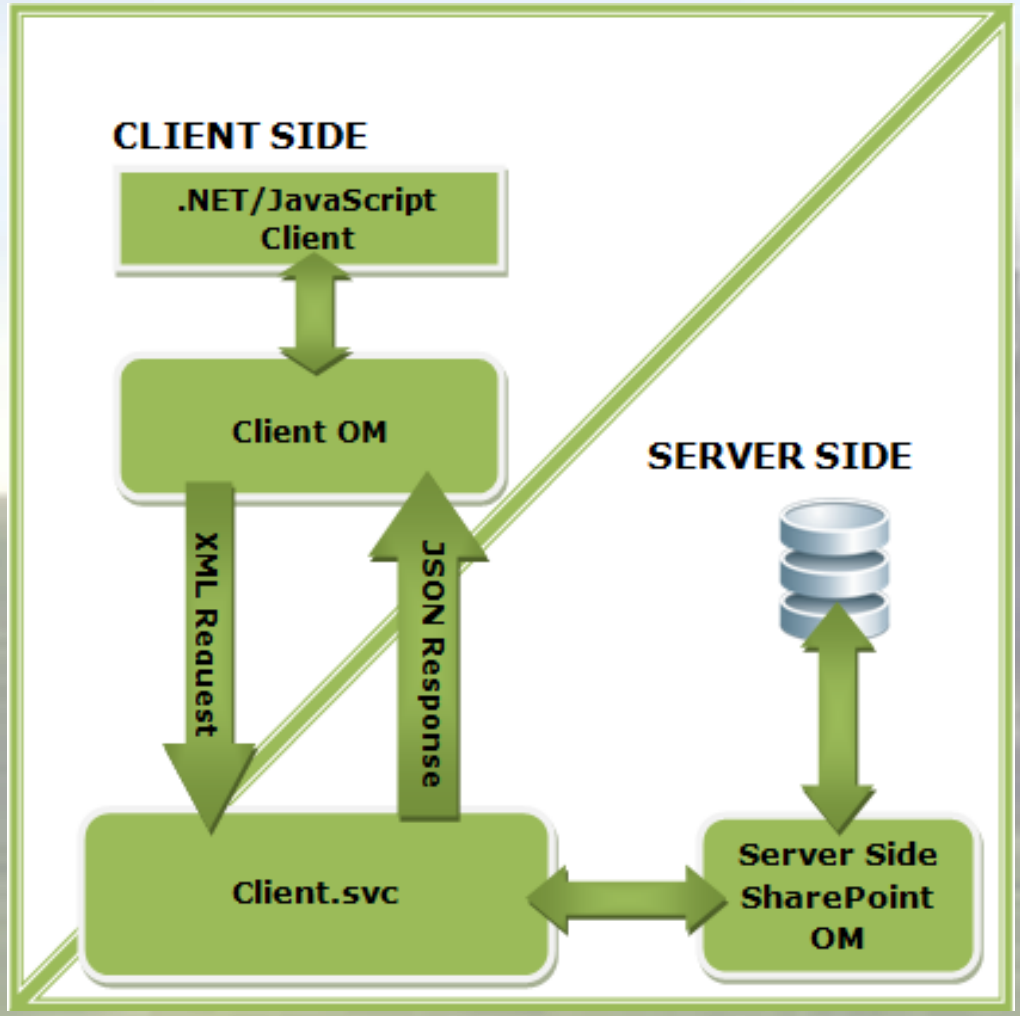
Agenda

- Introduction / Why COM?
- COM Architecture
- Coding Samples
- DEMO
  - .NET COM
- Questions

Intro to the SP 2010 COM

- Not enough web services in SP 2007
- Rather than create more services, COM provides the complete API
- COM provides a consistent development experience:
  - Windows Applications
  - ASP.NET web sites
  - Silverlight Applications
  - JavaScript, www client side scripting

COM Architecture



Assembly References

- SharePoint, Server Side
  - Microsoft.SharePoint (ISAPI)
- .NET clients
  - Microsoft.SharePoint.Client (ISAPI)
- Silverlight clients
  - Microsoft.SharePoint.Client.Silverlight (Layouts/clientbin)
- Javascript clients
  - SP.js & SP.Core.js (Layouts)

Comparable Objects

Microsoft.SharePoint	Client Object Models
SPContext	ClientContext
SPSite	Site
SPWeb	Web
SPList	List
SPListItem	ListItem
SPField	Field
SPFile	File

Starter Code

```
Using Microsoft.SharePoint.Client;
...
using (ClientContext context = new
  ClientContext("http://intranet"))
{
  Web web = context.Web;
  context.Load(web);
  context.ExecuteQuery();
  string title = web.Title;

  // ListCollection lists = web.Lists;
}
```

Iterating through Lists in a Web

```
using (ClientContext context = new
  ClientContext("http://intranet"))
{
  Web web = context.Web;
  context.Load(web);
  context.Load(web.Lists);
  context.ExecuteQuery();

  foreach(List list in web.Lists)
  {
    //do something
  }
}
```

Efficiencies… Don't be Lazy!

```
Web web = context.Web;
context.Load(web, wprop => wprop.Title));

ListCollection lists = web.Lists;
IEnumerable<List> filtered = context.
   LoadQuery(lists.Include(l=>l.Title));
context.ExecuteQuery();

foreach(List list in filtered)
{ }
```

Working with List Items

```
Web web = context.Web;
List list = context.Web.Lists.
          GetByTitle("List Title");

CamlQuery query = CamlQuery.CreateAllItemsQuery();

ListItemCollection items = lst.GetItems(query);
context.Load(items);
context.ExecuteQuery();

foreach (ListItem item in items)
{
  string title = item["Title"];
}
```

Efficencies with List Items

```
CamlQuery query = new CamlQuery();
query.ViewXml = "<View><Query><Where><Eq>
  <FieldRef Name='Title'/><Value
  Type='Text'>Phil</Value>
  </Eq></Where></Query></View>";

ListItemCollection items = list.GetItems(query);
context.Load(items, x => x.Include(
                        item => item["ID"],
                        item => item["Title"],
                        item => item.DisplayName));
```

Adding new List Items

```
List list = context.Web.Lists.
        GetByTitle("List Title");
context.Load(list);

ListItem newItem = list.AddItem(new
   ListItemCreationInformation());

newItem["Title"] = "My new item";
newItem.Update();
context.ExecuteQuery();
```

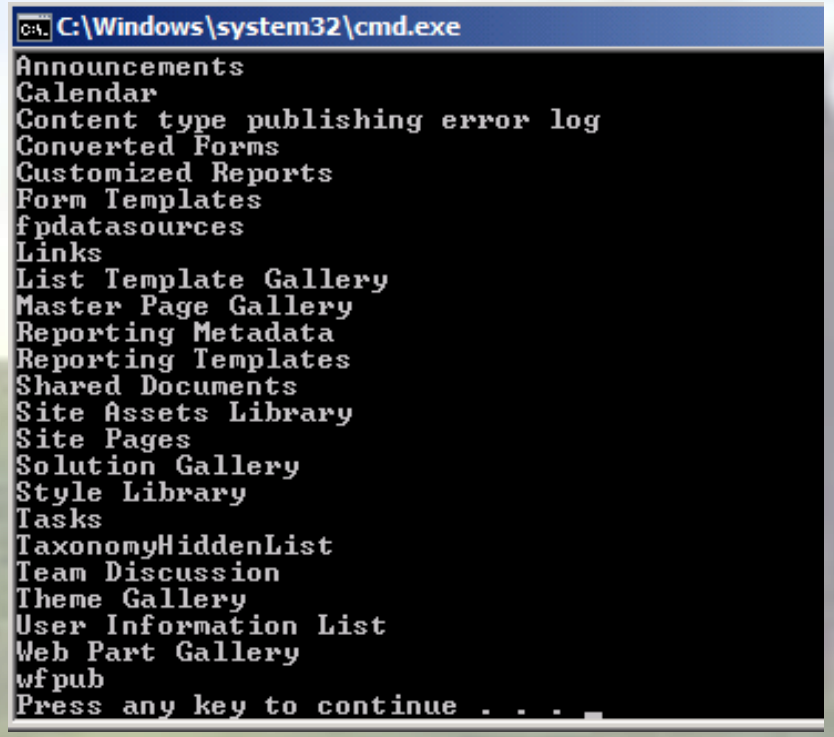
Silverlight & Asynchronous Calls

```
private void Button_Click(object sender, RoutedEventArgs e)
{
   // Load a bunch of stuff
   clientContext.ExecuteQueryAsync(success, failure);
}
private void success(object sender,
   ClientRequestSucceededEventArgs args)
{
   RunQuery runQuery= Run;
   this.Dispatcher.BeginInvoke(runQuery);
}
private delegate void RunQuery();
private void Run() {  /* do something */  }

private void failure(object sender,
   ClientRequestFailedEventArgs args) { /* do something */ }
```

.NET – COM Demo

- Build a Console (client) Application
- Render all the List Titles from a remote SharePoint site.
- Create a new list item in a remote SharePoint site.



SharePoint Happenings

QUESTIONS & COMMENTS

Phil Wicklund