ASP.NET MVC w/ Spark View Engine

Donn Felker
Principal – Agilevent

About

Donn Felker | Principal | Agilevent

donn@donnfelker.com [email]

blog.donnfelker.com [blog]

@donnfelker [Twitter]

Involvement

Twin Cities Dev. Guild | TwinCitiesDevelopersGuild.com

Twin Cities GiveCamp | TwinCitiesGiveCamp.org

Twin Cities Pragmatic Beer | TwinCitiesPragmaticBeer.com

Assumptions

You are Familiar with ASP.NET MVC

(This is **NOT** An Intro to ASP.NET MVC)

Under the Hood (view) Stays The Same

The Format

Code First

Setup Second

More Code

Why Another View Engine?

Brief History

- Created by Louis DeJardin
- Discussion on Phil Haacks Blog
- Topic: Code Based Repeaters
- <% … %> syntax is ugly

Spark was Born here

Peter
May 06, 2008
7:26 PM

re: Code Based Repeater for ASP.NET MVC

Arrr, all these <%..%> are sooo ugly... It's noise, they get in the way. That NVelocity example is not much prettier either. We need to think about it backwards. Forget what the current design view engine can or cannot do and let's start from scratch. What would we like?

Lou D
May 06, 2008
8:25 PM

re: Code Based Repeater for ASP.NET MVC

What if the few things you needed in a view language fit into the html seamlessly?

(crossing fingers about formatting)

```
<var i="0" />
<var css="new string[] {'row', 'row-alt'}" />
<table>
<for each="var hobby in Hobbies" i="i+1">
<tr class="$css[i%2];">
<td>$i;</td>
<td>$hobby.Title;</td>
<td>$Html.Function("arg","arg");</td>
</tr>
</for>
</table>
```

source

Wants

HTML Dominate the Flow

Clean

Rich Feature Set

Simple

Easy To Read

Easy To Test (Acceptance)

How is it clean and seamless?

MVC View Engine | Tag Soup

```
<%int rowIndex = 0; %>
<%foreach(AtwoodPost.Models.Page p in ViewData.Model.Pages){%>

    <tr class="<%=Html.GetRowClass(rowIndex) %>">
        <td><%=p.Title%></td>
        <td><%=p.Permalink%></td>
        <td><%=p.Created.ToLongDateString()%></td>
        <td><a href=#>Delete</a></td>

    </tr>
    <%rowIndex++; %>
<%} %>
```

Example Row Coloring Output:

Picture Source: codinghorror.com

Spark View Engine | HTML Dominates

```
<tr each="var p in Pages" class="alt?{pIndex % 2 == 0}">
    <td>${p.Title}</td>
    <td>${p.Permalink}</td>
    <td>${p.Created.ToLongDateString()}</td>
    <td><a href=#>Delete</a></td>
</tr>
```

Note: You get FREE Variables in Loops
- pIndex, pCount, pIsFirst, pIsLast

Spark Basics

Settings	web.config
View and layout files	*.spark
Output expressions	${…}
Output html*	!{…}
Inline code statements	#...;
Conditional attributes	foo="bar?{…}"

* Needed when <pages automaticencoding="true"/>

Accessing Data

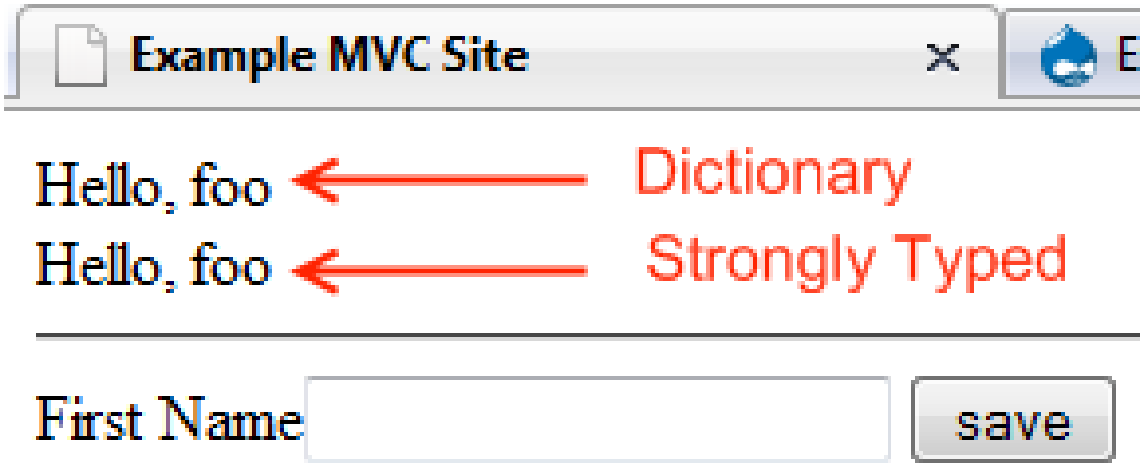
ViewData | Model

Accessing Data | ViewData

1. ViewData Dictionary
2. Strongly Typed Property Accessor

```
1  Hello, ${ViewData["name"]}<br/>
2
3  <viewdata name="string" />
4  Hello, ${name} <br/>
5
```

Output:



Accessing Data | Models

```
1  <viewdata model="SparkDemo.WebHost.Models.Customer">
2  Hello ${Model.FirstName} ${Model.LastName}
```

Outputs: Hello Foo Bar

Inline Code

Escape to C# with '#' operator

```
#Html.BeginForm("save", "home");
<label for="firstName" >First Name</label>
${Html.TextBox("firstName")}

<input type="submit" value="save" />
#Html.EndForm();
```

Outputs:

Inline Code Cont.

```
#var foo = "bar";

Classic<br/>
<%
//-- Classic web form method
%>
<% if(foo.Contains("ar")){ %>
Contains 'ar'<br/>
<% } %>

<hr/>
Spark<br/>
#// Output through the Text Writer
#if(foo.Contains("ar")) { this.Output.WriteLine("Contains 'ar'"); }
```

Outputs:

Classic
Contains 'ar'

Spark
Contains 'ar'

Demo

Accessing Data & Inline Code

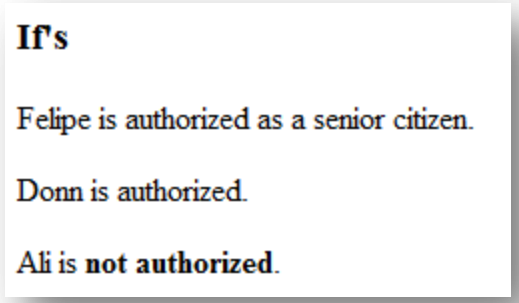
Conditionals

If | Test | Attributes

If / Else

```
<p each="var cust in Model">
    <if condition='cust.Age > 65'>
        ${cust.FirstName} is authorized as a senior citizen.
    </if>
    <else if='cust.Age > minAge'>
        ${cust.FirstName} is authorized.
    </else>
    <else>
        ${cust.FirstName} is <b>not authorized</b>.
    </else>
</p>
```

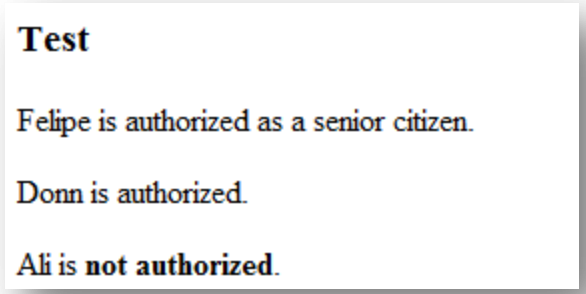
Output



Test

```
<h3>Test</h3>
<p each="var cust in Model">
    <test if='cust.Age > 65'>
        ${cust.FirstName} is authorized as a senior citizen.
    <else if='cust.Age > minAge'/>
        ${cust.FirstName} is authorized.
    <else/>
        ${cust.FirstName} is <b>not authorized</b>.
    </test>
</p>
```

Output

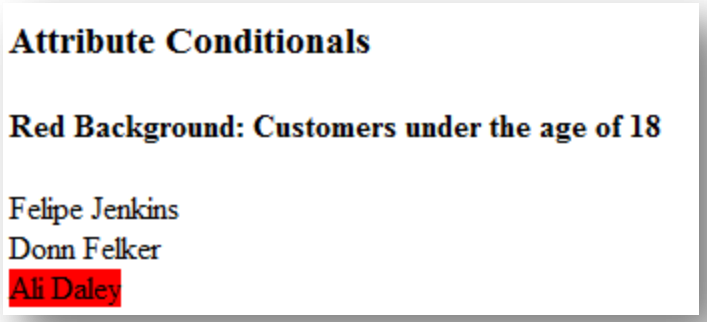


Attribute Conditionals

```
<h3>Attribute Conditionals</h3>
<h4>Red Background: Customers under the age of ${minAge} </h4>

<span each="var cust in Model"
        style="background-color:red?{cust.Age < minAge}">
    ${cust.FirstName} ${cust.LastName}<br/>
</span>
```

Output



Looping

each

Looping (for each)

```
<h3>Looping With Free Variables</h3>
<style>
tr.alt { background-color: #c4c4c4; }
</style>
<table width="500" cellspacing="0">
    <tr>
        <th>Row Id</th>
        <th>First Name</th>
        <th>Last Name</th>
        <th>Age</th>
        <th>Authorized</th>
    </tr>
    <tr each="var cust in Model" class="alt?{custIndex % 2 == 0}">
        <td>${custIndex}</td>
        <td>${cust.FirstName}
            ${custIsFirst ? "(First Record)" : String.Empty}
            ${custIsLast ? "(Last Record)" : String.Empty}</td>
        <td>${cust.LastName}</td>
        <td>${cust.Age}</td>
        <td><input type="checkbox"
                checked="?{cust.Age > 18}"
                disabled="?{cust.Age < 18}" /></td>
    </tr>
</table>
```

Looping (output)

Looping With Free Variables

Row Id	First Name	Last Name	Age	Authorized
0	Felipe (First Record)	Jenkins	72	☑
1	Donn	Felker	31	☑
2	Ali (Last Record)	Daley	16	☐

Demo

Conditionals & Looping

Setting up Spark

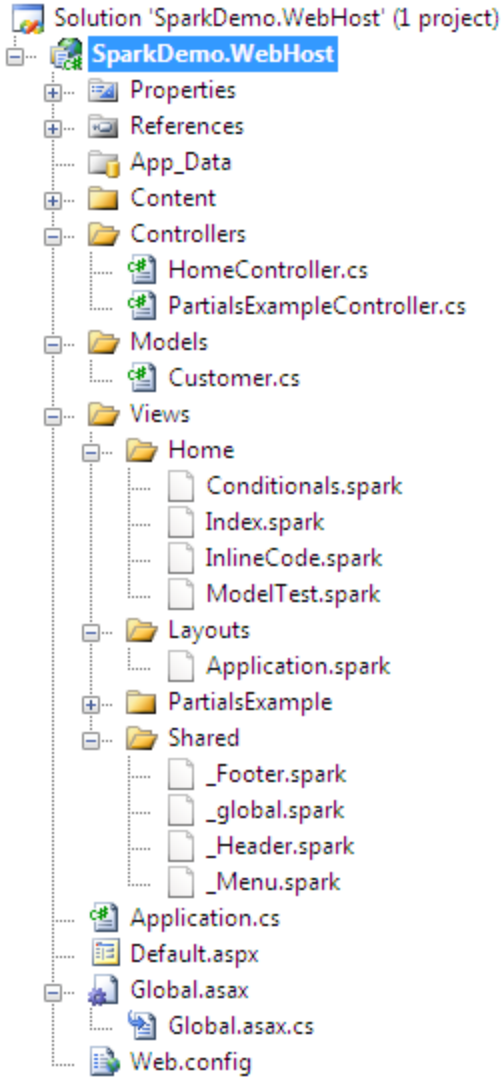
The Bits

1. Binaries sparkviewengine.com
   - Spark.dll + Spark.Web.Mvc.dll

2. Register Spark
   - SparkEngineStarter.RegisterViewEngine();

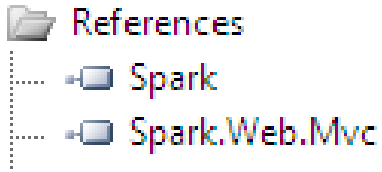
Migrating from MVC to Spark

Optional: Rename .as?x to .spark



File System

- ASP.NET MVC Application
- Add References



Registering Spark

```
public void RegisterViewEngines(ViewEngineCollection engines)
{
    if (engines == null) throw new ArgumentNullException("engines");

    SparkEngineStarter.RegisterViewEngine(engines);
}
```

Organizing Content

Content Locations & Partials

Named Content Areas

- Extensibility Points in your Web App
- Application.spark -

```
<html>
<head>
    <title>Example MVC Site</title>
     <link rel="stylesheet" href="~/C
     <use content="head" />
</head>

<body>
    <use content="view"/>
</body>
</html>
```

Named Content Cont

- The View:

```
<content:head>
 <!-- This will show in the header -->
</content:head>

Hello from Content Demo.

<content:head>
<!-- This will also show in the header -->
</content:head>
```

Named Content Output

- The output HTML:

```
<html>
<head>
        <title>Example MVC Site</title>
         <link rel="stylesheet" href="/Content/css/main.css
 <!-- This will show in the header -->
<!-- This will also show in the header -->
</head>

<body>

Hello from Content Demo.

</body>
</html>
```

Named Content - Once

- The “once” Attribute

```
<content:head>
  <script src="~/content/js/jquery-1.3.2.js" once="jquery"></script>

</content:head>

Hello from Content Demo # 2.

<content:head>
 <script src="~/content/js/jquery-1.3.2.js" once="jquery"     ></script>
</content:head>
```

Named Content - Once

- The “once” Attribute Output

```
<html>
<head>
        <title>Example MVC Site</title>
         <link rel="stylesheet" href="/Content/css/main.css" type="text/css"/>
  <script src="/content/js/jquery-1.3.2.js"></script>


</head>

<body>

Hello from Content Demo # 2.

</body>
</html>
```

Partials

- The “User Control” of Spark
- Located in /Views/Shared Folder
- Begin with “_” Example: _Menu.spark
- Accessed via element name in view. Example: <Menu />

Better to be seen than to be explained.

Demo

Partials

There's More

- Pre-Compilation
- Acceptance Testing
- Output Caching
- JavaScript View Result
- Pdf View Result (through iTextSharp)
- Modularity
- IronPython
- IronRuby
- Supports MonoRail
- Can be used as a General Templating Language
- Custom View Folders
- And much more …

(Examples are in the samples that come with the dll's)

Thank you

Donn Felker | Principal | Agilevent

donn@donnfelker.com [email]

blog.donnfelker.com [blog]

@donnfelker [Twitter]

Resources

Spark | SparkViewEngine.com

Screen Casts | DimeCasts.net

Louis DeJardin | whereslou.com