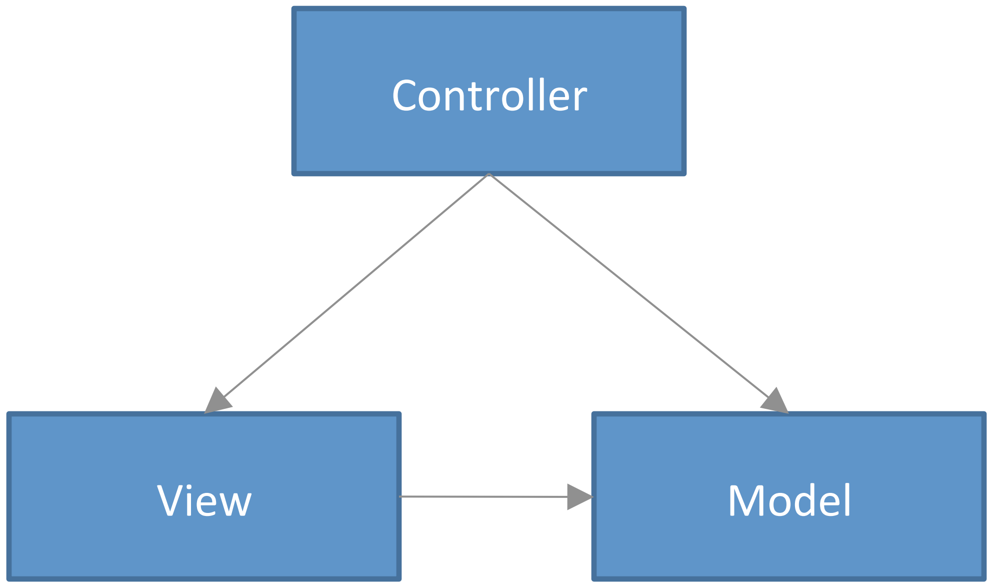
Introduction to ASP.NET MVC

kainos

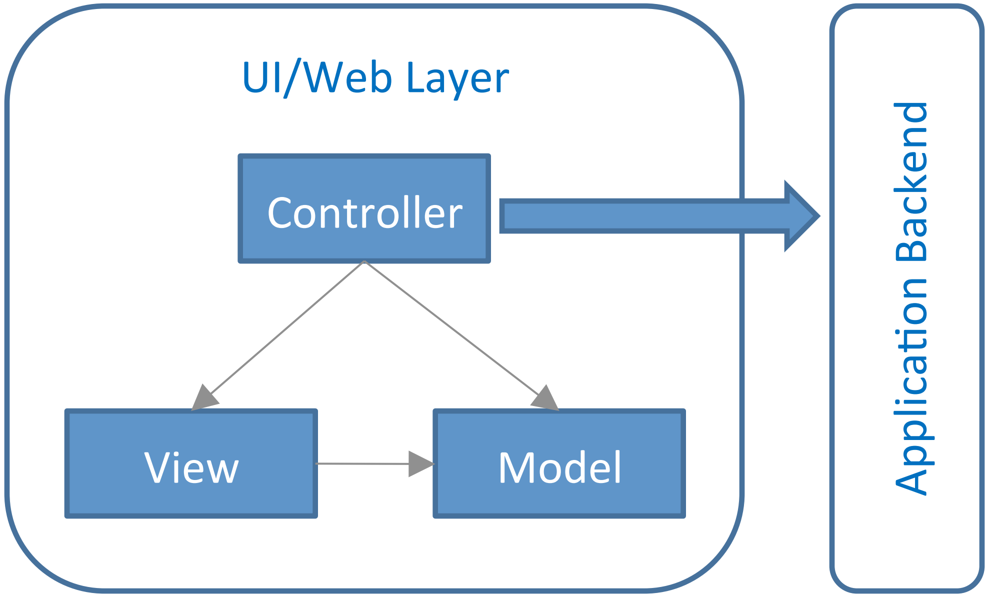
Overview

- MVC 101
- Overview of ASP.NET MVC
- My First Application Walkthrough
- ASP.NET MVC Brain Dump
- ASP.NET MVC 2.0 Features
- Questions

MVC



MVC in the UI Layer

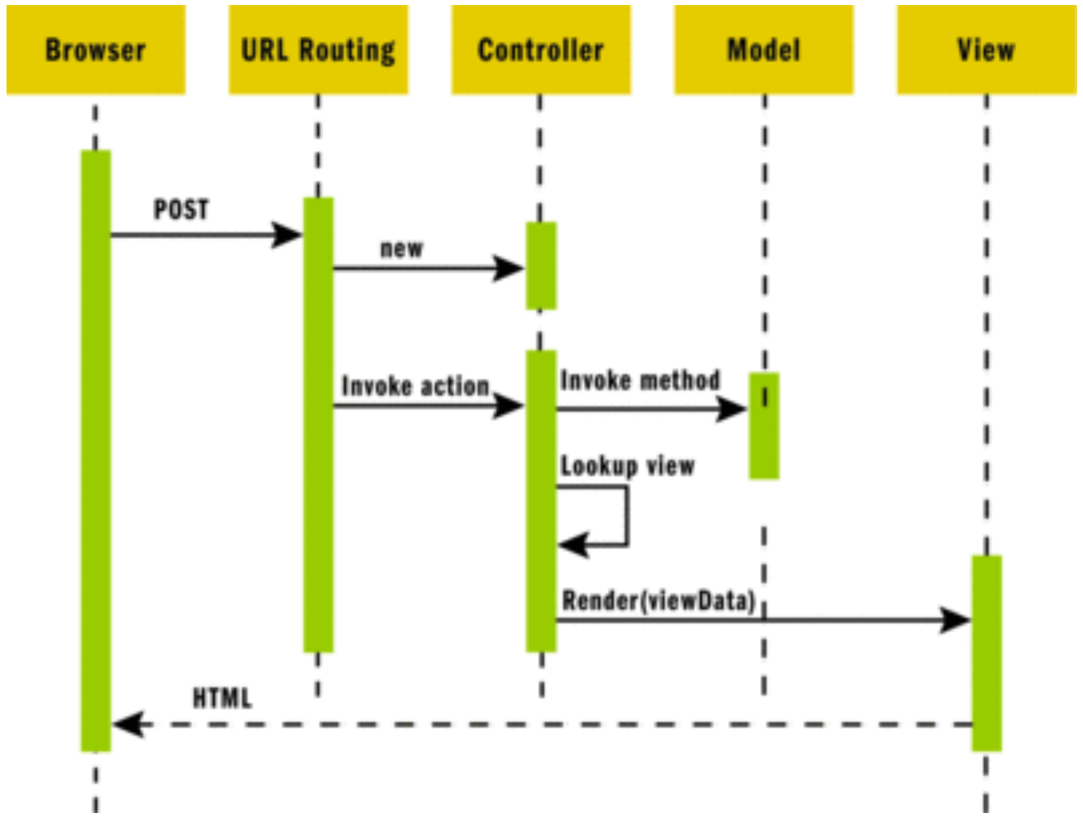


ASP.NET MVC Overview

- Framework built on the MVC pattern
- More mature than it seems
    - First CTP released in December 07
    - V1.0 released in March 09
    - V2.0 Preview 2 released Sep 09
- V2.0 bringing it' s A game
- http://www.asp.net/mvc

My First MVC Application

MVC Request Lifecycle



ASP.NET MVC Brain Dump

ASP.NET MVC

Alternative to WebForms

ASP.NET MVC

Fully extensible/pluggable

ASP.NET MVC

Better separation of concerns

ASP.NET MVC

More testable

ASP.NET MVC

Based on a proven design pattern

ASP.NET MVC

Less of an abstraction

ASP.NET MVC

Offers greater control over entire lifecycle

ASP.NET MVC

Cleaner URLs

ASP.NET MVC

No Postbacks or ViewState

ASP.NET MVC

Not for everyone (?)

ASP.NET MVC

Data Binding

ASP.NET MVC

Uses same ASP.NET Providers

ASP.NET MVC

Simplified communication mechanism

ASP.NET MVC

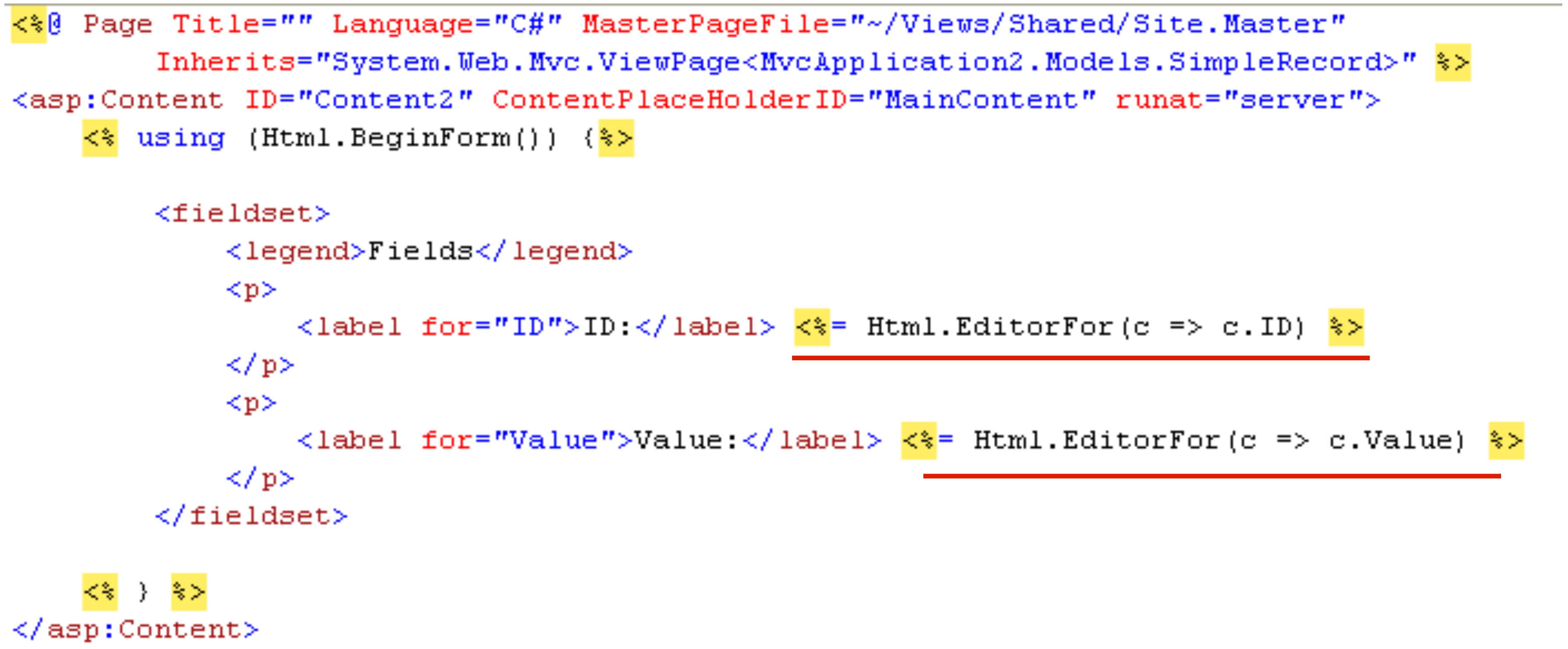
Offers "Coding by Convention"

ASP.NET MVC

Simplifies Cross Training

ASP.NET MVC 2 Features

Strongly Typed Helpers

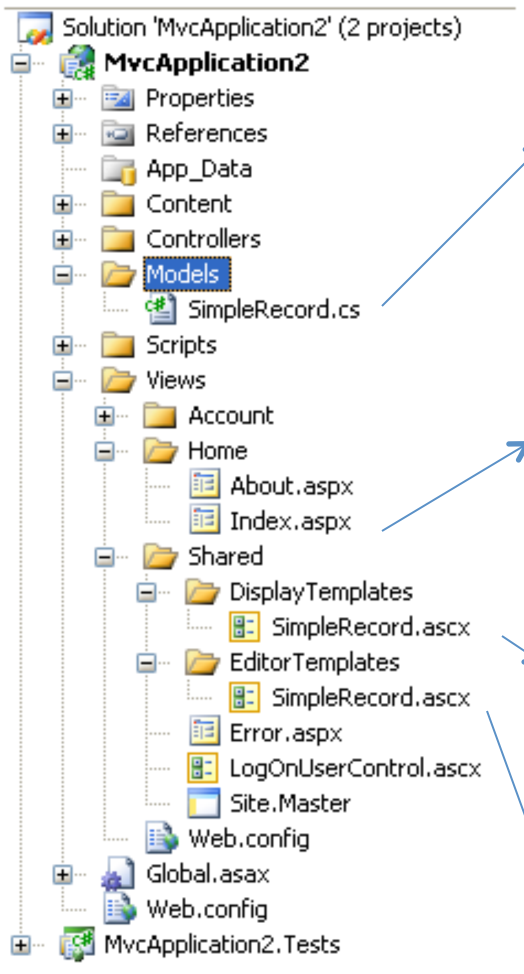


```
<%@ Page Title="" Language="C#" MasterPageFile="~/Views/Shared/Site.Master"
        Inherits="System.Web.Mvc.ViewPage<MvcApplication2.Models.SimpleRecord>" %>
<asp:Content ID="Content2" ContentPlaceHolderID="MainContent" runat="server">
    <% using (Html.BeginForm()) {%>

        <fieldset>
            <legend>Fields</legend>
            <p>
                <label for="ID">ID:</label> <%= Html.EditorFor(c => c.ID) %>
            </p>
            <p>
                <label for="Value">Value:</label> <%= Html.EditorFor(c => c.Value) %>
            </p>
        </fieldset>

    <% } %>
</asp:Content>
```

Templated Helpers



```
Solution 'MvcApplication2' (2 projects)
  MvcApplication2
    Properties
    References
    App_Data
    Content
    Controllers
    Models
      SimpleRecord.cs
    Scripts
    Views
      Account
      Home
        About.aspx
        Index.aspx
      Shared
        DisplayTemplates
          SimpleRecord.ascx
        EditorTemplates
          SimpleRecord.ascx
        Error.aspx
        LogOnUserControl.ascx
        Site.Master
      Web.config
    Global.asax
    Web.config
  MvcApplication2.Tests
```

```
namespace MvcApplication2.Models
{
    public class SimpleRecord
    {
        public int ID { get; set; }
        public string Value { get; set; }
    }
}
```

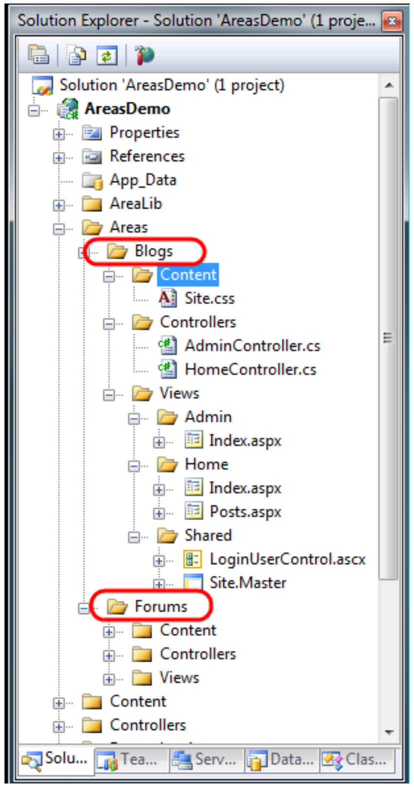
```
<h2>Display</h2>
<% foreach (var item in Model) { %>
    <%=Html.DisplayFor(c => item) %>
<% } %>

<h2>Edit</h2>
<% foreach (var item in Model) { %>
    <%=Html.EditorFor(c => item) %>
<% } %>
```

```
<%@ Control Language="C#" Inherits="System.Web.Mvc.ViewUserControl<MvcApplication2.Models.SimpleRecord>" %>
<p>
    <%=Html.DisplayFor(c => c.ID)%> : <%=Html.EditorFor(c => c.Value)%>
</p>
```

```
<%@ Control Language="C#" Inherits="System.Web.Mvc.ViewUserControl<MvcApplication2.Models.SimpleRecord>" %>
<p>
    <%=Html.DisplayFor(c => c.ID)%> : <%=Html.DisplayFor(c => c.Value)%>
</p>
```

Areas



Solution Explorer - Solution 'AreasDemo' (1 proje...

```
Solution 'AreasDemo' (1 project)
  AreasDemo
    Properties
    References
    App_Data
    AreaLib
    Areas
      Blogs
        Content
          Site.css
        Controllers
          AdminController.cs
          HomeController.cs
        Views
          Admin
            Index.aspx
          Home
            Index.aspx
            Posts.aspx
          Shared
            LoginUserControl.ascx
            Site.Master
      Forums
        Content
        Controllers
        Views
    Content
    Controllers
```

Client Side Validation

```
namespace MvcApplication2.Models
{
    public class SimpleRecord
    {
        [Required]
        [Range(0, 100, ErrorMessage="ID must be between 0 and 100")]
        public int ID { get; set; }

        [Required(ErrorMessage=" You must have a value you idiot!")]
        [StringLength(25)]
        public string Value { get; set; }
    }
}
```

Annotated Model

```
[HttpGet]
public ActionResult CreateRecord()
{
    return View();
}

[HttpPost]
public ActionResult CreateRecord(SimpleRecord record)
{
    if (ModelState.IsValid)
    {
        return RedirectToAction("Index");
    }
    else
    {
        return View(record);
    }
}
```

Controller Actions

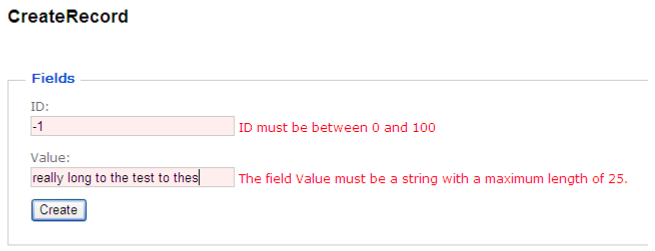
```
<script src="../../Scripts/jquery-1.3.2.min.js" type="text/javascript"></script>
<script src="../../Scripts/jquery.validate.min.js" type="text/javascript"></script>
<script src="../../Scripts/MicrosoftMvcJQueryValidation.js" type="text/javascript"></script>
<%Html.EnableClientValidation();%>

<h2>CreateRecord</h2>

<%= Html.ValidationSummary("Create was unsuccessful. Please correct the errors and try again.") %>
<% using (Html.BeginForm()) {%>

    <fieldset>
        <legend>Fields</legend>
        <p>
            <label for="ID">ID:</label>
            <%= Html.TextBox("ID") %>
            <%= Html.ValidationMessage("ID", "*") %>
        </p>
        <p>
            <label for="Value">Value:</label>
            <%= Html.TextBox("Value") %>
            <%= Html.ValidationMessage("Value", "*") %>
        </p>
        <p>
            <input type="submit" value="Create" />
        </p>
    </fieldset>
```

View



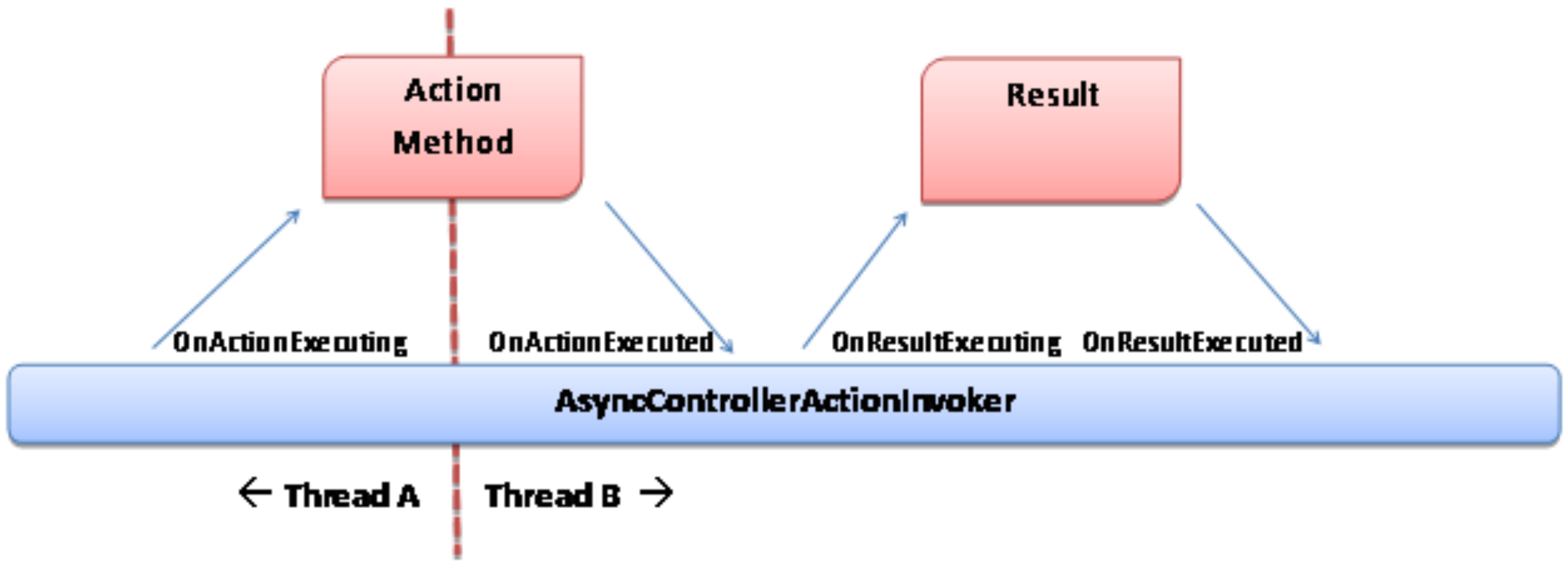
CreateRecord

Fields

ID:
-1 ID must be between 0 and 100

Value:
really long to the test to thes The field Value must be a string with a maximum length of 25.

Async Controller Actions



Questions?