

Inside ASP.NET MVC

Now with added
Preview 3

David Wheeler
INSIDE ASP.NET MVC

© 2008 David Wheeler. All Rights Reserved.

Session agenda

- BRIEF refresher on ASP.NET architecture
- ASP.NET MVC defined
- A series of demos that unpick ASP.NET MVC
 - Introduction
 - Routing
 - Controllers and Views
 - Forms
 - Extensibility
 - Mocking and testing

© 2008 David Wheeler. All Rights Reserved.

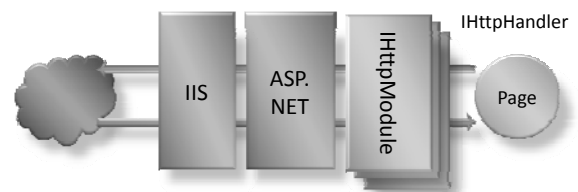
Disclaimer

- This is based on ASP.NET MVC Preview 3
 - Only had it installed since Saturday!
 - All, some or none of this might change before release, which may, or may not, ever happen
- You have been warned!



© 2008 David Wheeler. All Rights Reserved.

"Classic" ASP.NET Processing pipeline



© 2008 David Wheeler. All Rights Reserved.

"Classic" ASP.NET Main features

- High level abstraction over HTML / HTTP
- Simplified state management
 - ViewState and the post-back model
- Control model
 - Data binding
 - Simple event-driven mechanism
- Simple Page Controller pattern
- And lots more as well...

© 2008 David Wheeler. All Rights Reserved.

"Classic" ASP.NET (Some of) the downsides

- Sub-optimal URLs
 - blog.aspx?date=21032008
- form runat="server"
- ViewState
- Hard to test
 - All sorts of code in the page
 - Requirement to test with an HttpContext
 - Typically involves using Cassini / IIS

© 2008 David Wheeler. All Rights Reserved.

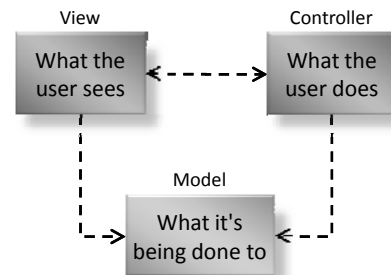
Inside ASP.NET MVC

ASP.NET MVC defined

- A new project-type for VS 2008
- A new routing mechanism
 - Applicable not just to ASP.NET MVC
- Easier to write using TDD
 - Clean separation of concerns
 - Mockable intrinsics (e.g. HttpContextBase)
- NOT a replacement for existing Web Forms
 - Feel free to completely ignore ASP.NET MVC

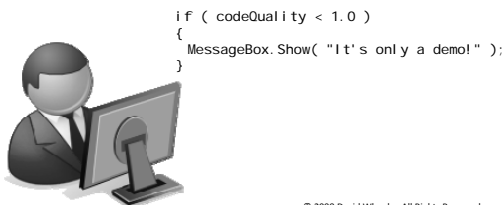
© 2008 David Wheeler. All Rights Reserved.

MVC defined



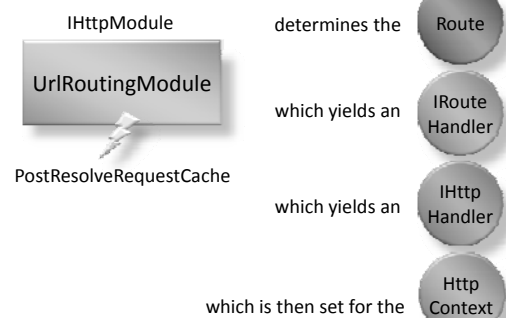
© 2008 David Wheeler. All Rights Reserved.

DEMO Hello, ASP.NET MVC



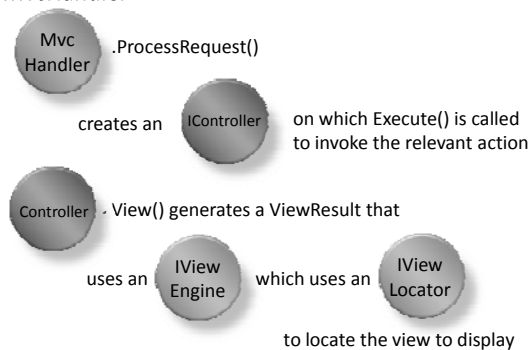
© 2008 David Wheeler. All Rights Reserved.

There's no magic The routing mechanism



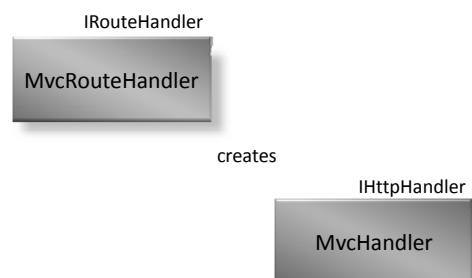
© 2008 David Wheeler. All Rights Reserved.

There's no magic MvcHandler



© 2008 David Wheeler. All Rights Reserved.

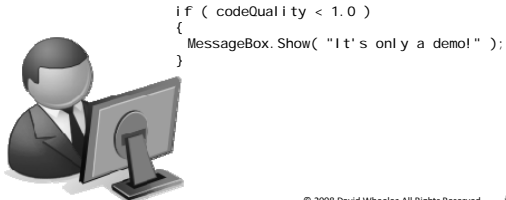
Routing and MVC The key players



© 2008 David Wheeler. All Rights Reserved.

Inside ASP.NET MVC

DEMO Examining the request pipeline



© 2008 David Wheeler. All Rights Reserved.

Routing

- Routing provides "clean" URLs
 - URL is mapped to a route handler
- Extra level of indirection
 - Handlers can be changed without impacting URL
 - URL can be changed without impacting handler
- Enables support for multilingual URLs

© 2008 David Wheeler. All Rights Reserved.

The routing table A basic route mapping

```
public class GlobalApplication : System.Web.HttpApplication
{
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");
        routes.MapRoute(
            "Default",
            "{controller}/{action}/{id}",
            new { controller = "Home", action = "Index", id = "" } );
    }

    protected void Application_Start(object sender, EventArgs e)
    {
        RegisterRoutes(RouteTable.Routes);
    }
}
```

© 2008 David Wheeler. All Rights Reserved.

The routing table More complex routes

```
public class GlobalApplication : System.Web.HttpApplication
{
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.MapRoute (
            // Name of the route
            "Weather",
            // Url to match against
            "weather/{year}/{month}/{day}",
            // new Controller
            new { controller="Weather",
                action="Display",
                year = "2008", month="01", day = "01" },
            // constraints
            new { year = @"^\d{4}", month=@"^\d{2}",
                day = @"^\d{2}" } )
    }
    ... // other route mappings
}
```

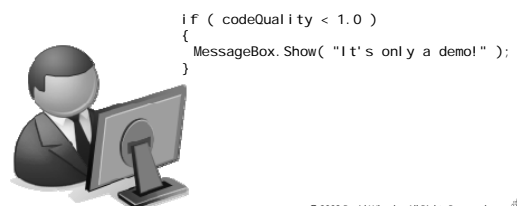
© 2008 David Wheeler. All Rights Reserved.

Route constraints Three basic options

- variable="regexExpression"
 - Use a regular expression to match the variable
- httpMethod =
new HttpMethodConstraint("GET", "PUT");
 - Apply a constraint on the HTTP verb
- variable=anObject
 - where anObject implements IRouteConstraint
 - Allows you to execute code in a constraint

© 2008 David Wheeler. All Rights Reserved.

DEMO Routing



© 2008 David Wheeler. All Rights Reserved.

Inside ASP.NET MVC

Controllers

- Just a class that implements `Controller`
 - Typically derive from the provided `Controller` class
- Public methods are the *action* methods
 - Invoked via the routing mechanism
 - Apply `[NonAction]` attribute to turn this off
- Action method prepares data and shows a view
 - Controller sets the data that the view uses
 - Use the `View` method to display the view

© 2008 David Wheeler. All Rights Reserved.

Controllers

Core action functionality

- Actions methods should return an `ActionResult`
 - `ViewResult` - if you want to show a View
 - `ContentResult` - directly returning content
 - `RedirectResult` - to redirect to a new URL
 - `RedirectToRouteResult` - redirect to route
 - `EmptyResult` - null object pattern
 - `JsonResult` - returning a blob of Jason

© 2008 David Wheeler. All Rights Reserved.

Controllers

Funky feature: `ActionFilterAttribute`

- Abstract attribute class
 - Derive to make a concrete type
- Apply concrete type to action methods
 - ASP.NET MVC will automatically call it
- Useful for AOP features
 - Logging, transactions, etc.

© 2008 David Wheeler. All Rights Reserved.

Views

- ASP.NET MVC provides a `WebFormViewEngine`
 - Allows you to use `.aspx`, `.ascx` and `.master` for view
- `ViewPage` derives from `System.Web.UI.Page`
 - Largely minus the server-side infrastructure
 - Ignore post-back, etc.
- Views can do anything you like
 - Render HTML, XAML, produce images, etc.

© 2008 David Wheeler. All Rights Reserved.

Views

Views need data!

- Controller obtains it from the model
 - (or even just passes in the model)
- Controller sets data via `ViewData` dictionary
 - e.g. `ViewData["Title"] = "MVC Rocks";`
- Strongly typed data passed in `Model` property
 - View declared using `ViewPage<T>`
 - `T` defines the type of the `ViewData.Model` property

© 2008 David Wheeler. All Rights Reserved.

DEMO

Controllers and Views



```
if ( codeQuality < 1.0 )  
{  
    MessageBox.Show( "It's only a demo!" );  
}
```

© 2008 David Wheeler. All Rights Reserved.

Inside ASP.NET MVC

URLs, forms and other helpers

- Two major items to consider in a site
 - How to implement hyperlinks and forms
- ASP.NET MVC provides assistance here
 - Html helper class
 - e.g. `Html.ActionLink`, `Html.Form`, `Html.TextBox`, etc.
 - Url helper
 - e.g. `Url.Action`
 - `BindingHelperExtensions.UpdateFrom`
 - Reading data from form variables

© 2008 David Wheeler. All Rights Reserved.

DEMO

Forms and helpers



```
if ( codeQual i ty < 1.0 )  
{  
    MessageBox.Show( "It's only a demo!" );  
}
```

© 2008 David Wheeler. All Rights Reserved.

Extensibility

- Just about everything is extensible
 - Custom routing handlers (`IRouteHandler`)
 - Custom handlers (`IHttpHandler`)
 - Custom controllers (`Controller`, `ControllerFactory`)
 - Custom views (`ViewEngine`)
 - Custom view locator (`ViewLocator`)
 - Custom views (`ViewDataContainer`)

© 2008 David Wheeler. All Rights Reserved.

DEMO

Extensibility



```
if ( codeQual i ty < 1.0 )  
{  
    MessageBox.Show( "It's only a demo!" );  
}
```

© 2008 David Wheeler. All Rights Reserved.

Mocking and testing

- `HttpContext` et al were always hard to mock
- ASP.NET MVC introduces mockable versions
 - `HttpContextBase`, `HttpRequestBase`, etc.
- Enables testing from the command line
 - No ASP.NET runtime needed

© 2008 David Wheeler. All Rights Reserved.

DEMO

Mocking and testing



```
if ( codeQual i ty < 1.0 )  
{  
    MessageBox.Show( "It's only a demo!" );  
}
```

© 2008 David Wheeler. All Rights Reserved.

Inside ASP.NET MVC

Summary

- ASP.NET MVC does NOT replace Web Forms
 - Both approaches are valid moving forward
 - Both can utilise features such as routing
- ASP.NET MVC is better suited to TDD
 - Greater separation of concerns
- ASP.NET MVC utilises many ASP.NET features
 - Master pages, role-based security, etc.
- ASP.NET MVC is not for everyone!

© 2008 David Wheeler. All Rights Reserved.



Thanks for coming

Q & A

© 2008 David Wheeler. All Rights Reserved.

