Webstack of ♥

HTML5

Knockout

Web API

SignalR

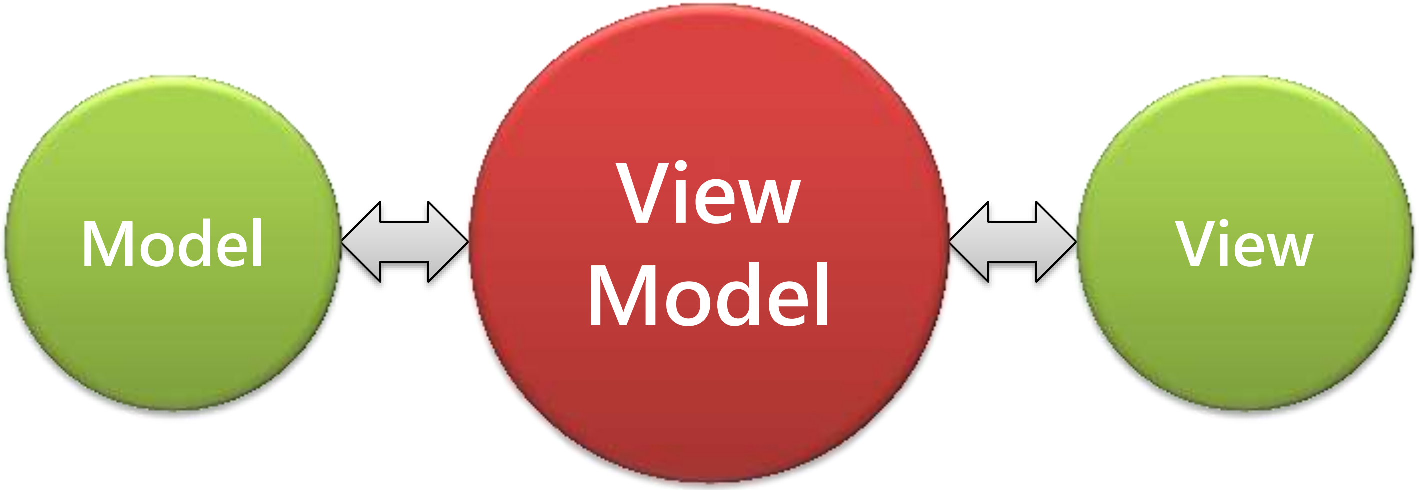
@bradwilson

http://bradwilson.typepad.com/

Demo

UI Mockup from the “Designer”

Knockout.



Model

View Model

View

Business Layer

Knockout (Observables)

HTML Markup

Declarative

```
<div>
  Hello, <strong>Brad</strong>
</div>
```

Declarative

```
<div>
  Hello,
  <strong data-bind="text: name" />
</div>
```

Knockout.

Iterative

```
<ul>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
<ul>
```

Iterative

```
<ul data-bind="foreach: items">
  <li data-bind="text: name"></li>
<ul>
```

Eventing

```
<input
  type="button"
  onclick="someJavaScript(thisObj)" />
```

Eventing

```
<input
  type="button"
  data-bind="click: modelFunction" />
```

Templating

```
<div>
  <span data-bind="text: buyer().firstName" />
  <span data-bind="text: buyer().lastName" />
</div>
<div>
  <span data-bind="text: seller().firstName" />
  <span data-bind="text: seller().lastName" />
</div>
```

Knockout.

Templating

```
<script type="text/html" id="fullName">
  <span data-bind="text: firstName" />
  <span data-bind="text: lastName" />
</script>


<div data-bind="template:
    { name='fullName', data: buyer }" />
<div data-bind="template:
    { name='fullName', data: seller }" />
```

Knockout.

Observables

- View models with simple values are read once
- View models with observables get live updates

```
ko.observable(value)
ko.observableArray([...])
ko.computed(function()  { ... })
ko.applyBindings(viewModel)
```

Demo

Convert Mockup to Knockout

ASP.NET Web API

Elevator Pitch

MVC excels at processing form data and returning HTML.

Web API excels at processing and returning structured data like JSON or XML.

When you want to do both, use both.

ASP.NET Web API

Mix in a Little of the Old…

- Routing
- Controllers and actions
- Filters
- Model binding
- Dependency injection support

ASP.NET Web API

…With a Little of the New

- Dispatch based on HTTP verb
- Parameter binders
- Formatters
- Introspection
- Async from top to bottom (using `Task<T>`)
- Pluggable hosting layer

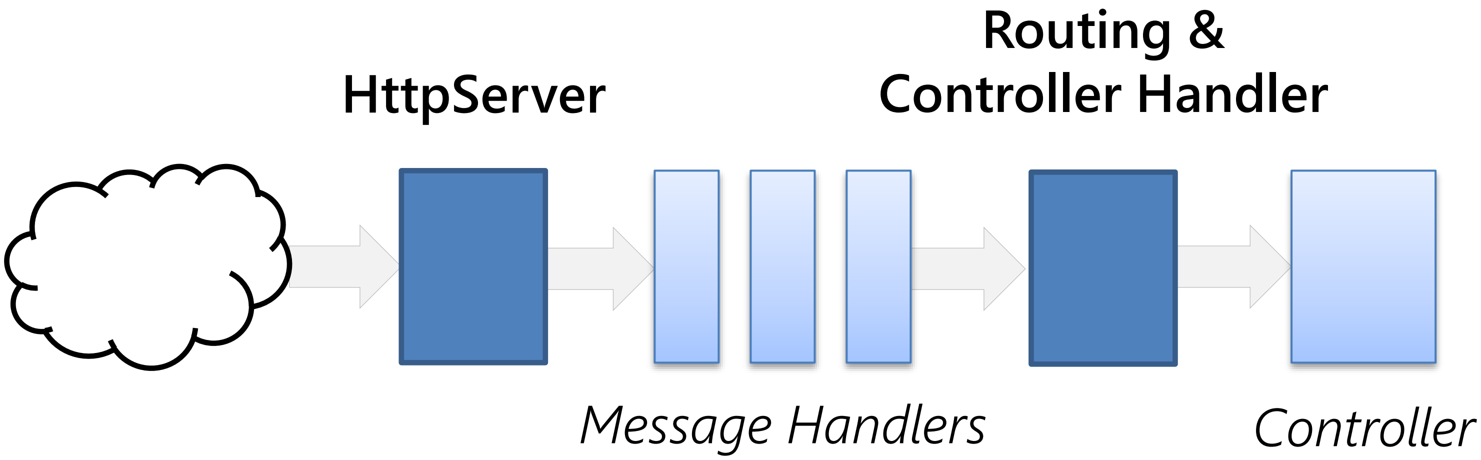
ASP.NET Web API

Inspired by System.Net.Http

- HttpRequestMessage in
- HttpResponseMessage out
- HttpMessageHandler in-between
- *No thread affinity, no static state**

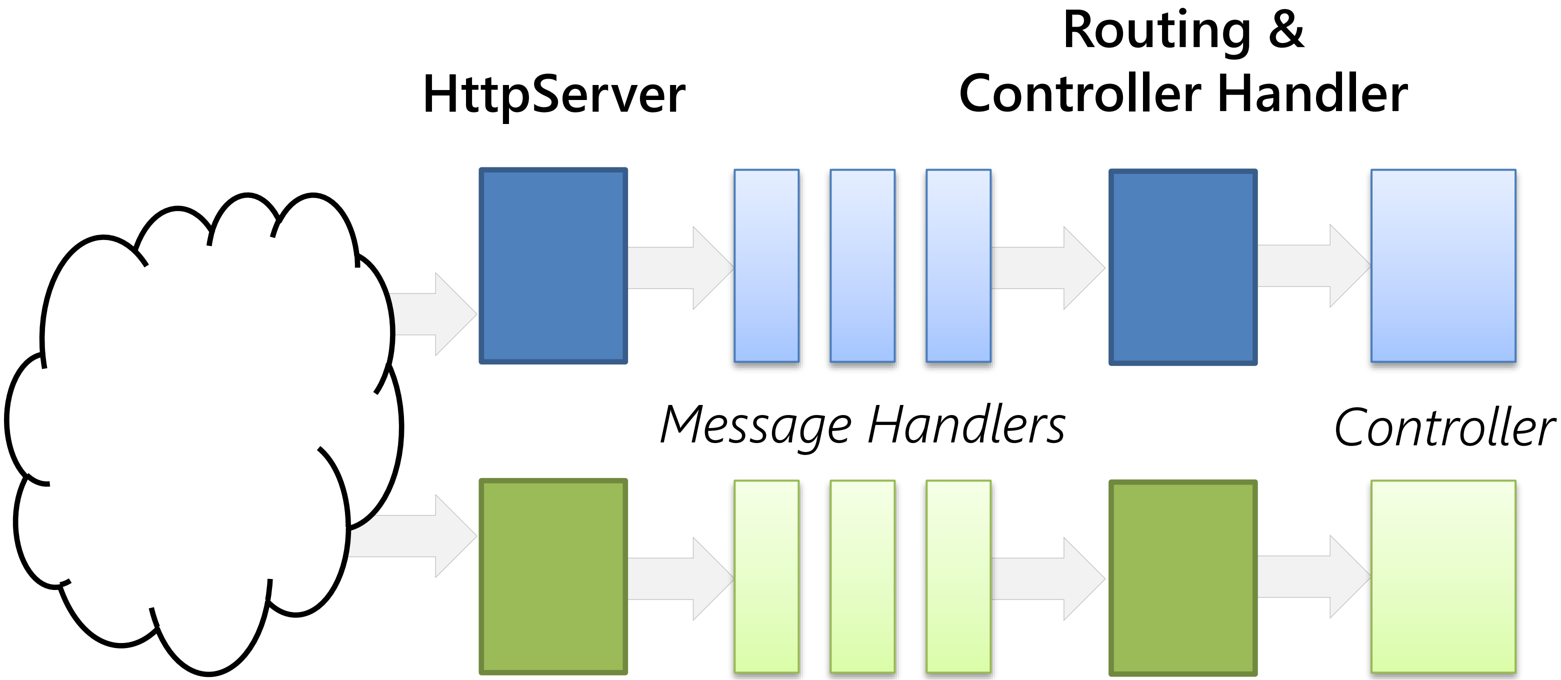
ASP.NET Web API

Dispatch (Self Host)



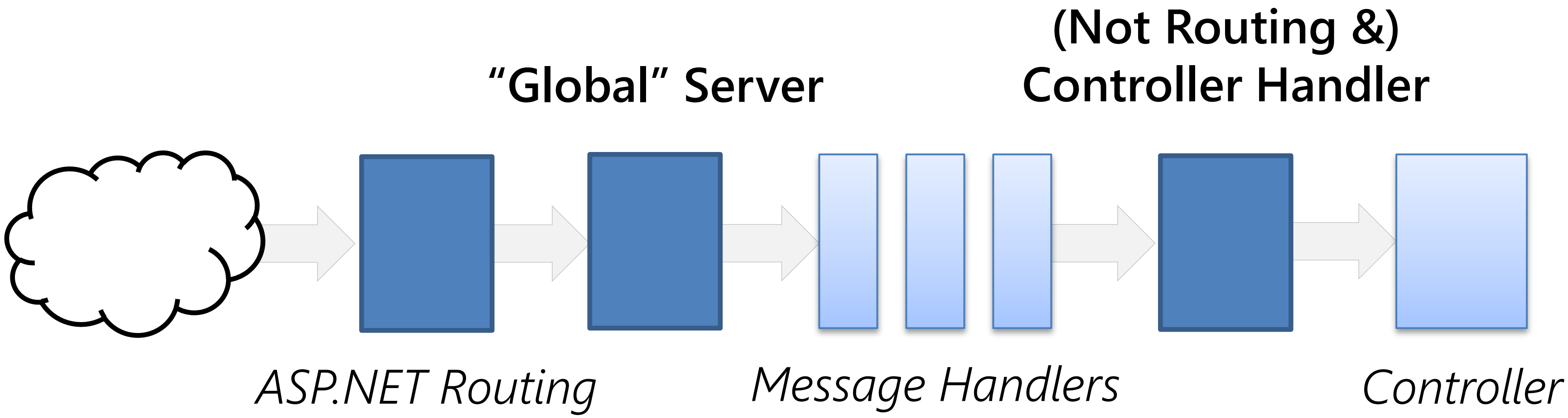
ASP.NET Web API

Dispatch (Self Host)



ASP.NET Web API

Dispatch (Web Host)



ASP.NET Web API

Built as (and with) Open Source

http://aspnetwebstack.codeplex.com

- JSON.NET
- DotNetOpenAuth
- *Tools, too: WiX, xUnit.net, Moq, StyleCop*

Demo

Add Web API for Persistence

SignalR

Connection Mechanisms

- Web sockets
- Server-sent events
- Forever frames
- Long polling

SignalR

Two levels of abstraction

- Persistent connections
  - Send/receive discrete strings
- Hubs
  - RPC-style messages

Server library: .NET

Client libraries: .NET, JavaScript, + others

SignalR

Persistent Connection

- Connection events
  - Connected / Reconnected
  - Disconnected
- Information distribution
  - Send to specific client
  - Broadcast to all clients

SignalR

Hubs (Server, inbound)

```
public class MyHub : Hub {
  public string Echo(string value) {
    return value + "!";
  }
}
```

SignalR

Hubs (client, outbound)

```
var hub = $.connection.myHub;
var result = hub.echo("value");
// result is "value!"
```

SignalR

Hubs (client, inbound)

```
var hub = $.connection.myHub;

hub.sayItLoud = function(message) {
  alert(message);
};

hub.tellAll("Hello, world!");
```

SignalR

Hubs (server, outbound)

```
public class MyHub : Hub {
  public void TellAll(string message) {
    Clients.sayItLoud(message);
    // Caller.sayItLoud(message);
    // Clients[id].sayItLoud(message);
  }
}
```

Demo

Live Collaboration with SignalR

Thank you!

github.com/bradwilson/WebstackOfLove
bradwilson.typepad.com/blog/talks.html