

DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

Build Rich Data Web APIs with ASP.NET

Brian Noyes

CTO & Co-founder, Solliance
brian.noyes@solliance.net, @briannoyes

Demos: <http://tinyurl.com/devinteuDataServicesDemos>



Tweet this talk

#DEVintersectionEU@briannoyes

DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

Thank You Sponsors!

 GitHub

 DevExpress

 JetBrains

 SDN
SOFTWARE DEVELOPMENT NETWORK
www.sdn.nl

About Brian Noyes

www.solliance.net

CTO and Co-founder, Solliance
www.solliance.net

Microsoft Regional Director

Microsoft MVP

Microsoft
Regional Director



Pluralsight author
www.pluralsight.com

pluralsight

Web API Insider, Windows Azure Insider,
Window Store App Insider, C#/VB Insider

 brian.noyes@solliance.net

 [@briannoyes](https://twitter.com/briannoyes)

 <http://briannoyes.net>

Agenda

- Data Services Overview
- Building CRUD Data Services with Web API
- OData Protocol
- Building OData Services with Web API
- Consuming OData

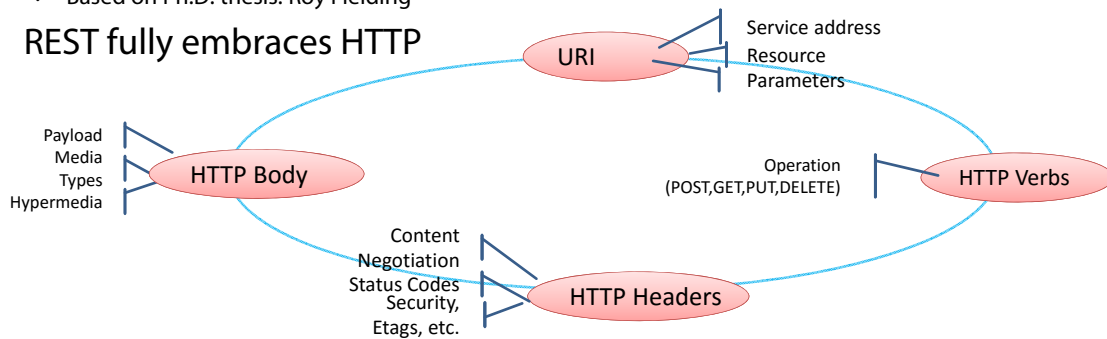
Data Services Options

- ASP.NET Web API
 - CRUD services
 - OData data services
- WCF Data Services
 - OData services
- WCF
 - CRUD-centric RPC SOAP services
 - CRUD-centric WebHttp services



REST

- Representational State Transfer
- REST is an architectural style
 - Based on Ph.D. thesis: Roy Fielding
- REST fully embraces HTTP



ASP.NET Web API Overview

- Platform for building HTTP Web Services (Web APIs)
- First released with ASP.NET MVC 4
- Web API 2 released with .NET 4.5
 - Separate HTTP request processing stack from MVC
 - OWIN
- ASP.NET 5 changes a lot
 - Processing stacks merge again
 - Web APIs just become a particular style of MVC controller - Same base class Controller
 - Ability to host on full.NET or .NET Core (beta)
- Makes it easy to build services with many architectural styles
 - Simple RPC services
 - CRUD services
 - REST services
 - OData services



ASP.NET Web API Overview

- Services are Controllers
 - ApiController base class (ASP.NET 4 thru 4.6)
 - Controller base class (ASP.NET 5)
- Web API features
 - Routing
 - Model Binding
 - Action Filters



ASP.NET Web API Configuration

- No config file settings needed
- `HttpConfiguration` class
 - Associated with the ASP.NET web application instance
 - Accessible from `Global.asax` code behind in ASP.NET 4.6 and earlier projects
 - Create in your `Startup.cs` for OWIN / ASP.NET 5
 - Calls `WebApiConfig.Register`
 - Defaults are good enough for basic Web APIs
 - Can plug in formatters, filters, message handlers and other custom extensibility objects through this class



ASP.NET Web API Configuration

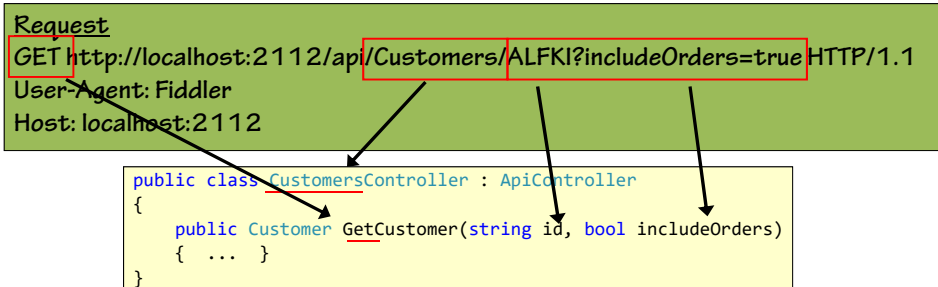
- `WebAPIConfig`

```
public static class WebApiConfig
{
    public static void Register(HttpConfiguration config)
    {
        config.Routes.MapHttpRoute(
            name: "DefaultApi",
            routeTemplate: "api/{controller}/{id}",
            defaults: new { id = RouteParameter.Optional }
        );
    }
}
```



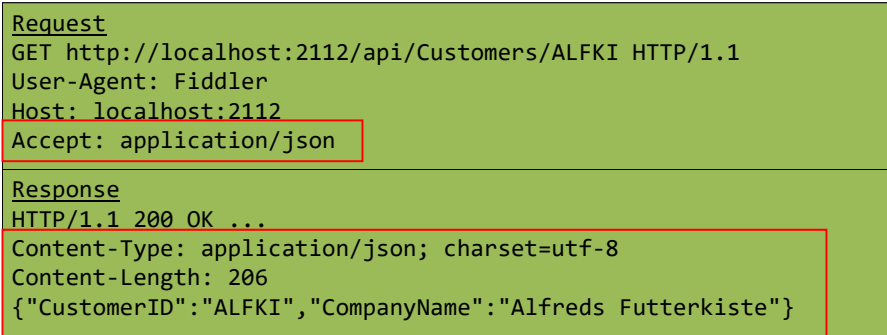
ASP.NET Web API Overview

- Convention over configuration
 - Maps URIs to controllers
 - Maps HTTP verbs to methods / actions
 - Maps URI / query string parameters to method parameters
- Can override verb mapping with attributes
- Can override all with attribute routing



ASP.NET Web API Overview

- Content negotiation
 - Based off HTTP Accept / Content-Type headers
 - JSON / XML formatters out of the box
 - OData formatter through NuGet
 - Can plug in custom formatters



OData

- Industry OASIS open standard protocol (www.odata.org)
 - Led by Microsoft, IBM, SAP, many other big companies
- Expose data over HTTP services for query or update
- REST-based services
 - Hypermedia links for related resources and actions
 - Includes metadata for client code generation
- Current version: 4.0



OData

- OData Query Syntax: URL syntax for expressing queries
 - Data service URI
 - Entity set name
 - Navigation property
 - Operators and functions
- OData Formatting: ATOM Publishing Protocol or JSON formatting
 - application/atom+xml or application/json
 - JSON Verbose vs JSON Light
- HTTP Verbs to express operation
 - GET, POST, PUT, DELETE, PATCH, MERGE



OData Operators

- Query string parameters
 - \$filter, \$orderby, \$skip, \$top, \$inlinecount
 - \$select, \$expand
- Filter operators
 - eq, ne, gt, lt, ge, le, and, or, not
 - add, sub, mul, div, mod, ()
- String functions
 - length, substring, indexof, etc.
- Date functions
 - year, month, day, minute, second
- Math functions
 - round, ceiling, floor
- Type functions
 - IsOf, Cast



Supporting OData Queries

- Can support just the OData query syntax for client driven queries on your resource collections
- Add ASP.NET Web API OData NuGet
- [EnableQuery] attribute
- Return IEnumerable<T> or IQueryable<T>
 - IEnumerable<T> applies filter in server stack after controller produces collection
 - IQueryable<T> merges server produced expression tree and the client query expression, then executes on the server (through provider)



Supporting OData Formats

- Atom Publishing Protocol or JSON
- Need an Entity Data Model for your exposed resource collections/objects
 - Implicit or explicit
- Need OData routing conventions for queries, navigation links, metadata, etc.



Consuming OData

- .NET:
 - WCF Data Services Client Library
 - Add Service Reference
 - DataServiceUtil.exe
 - .NET OData Client v4 code generator
- Javascript:
 - JQuery AJAX
 - DataJS library
 - BreezeJS library
 - JayData
- Tools
 - OData Explorer
 - LINQPad
 - Fiddler



What's Missing in Web API OData?

- Web API Batch update handling
 - No transaction around batch
 - Cannot guarantee correct ordering of operations
- Metadata for relationships
 - Missing some information on FK relations
 - Missing server type info for correlating entities
 - Workaround: Use Breeze EdmBuilder instead of ODataConventionModelBuilder
 - <http://www.getbreezenow.com/documentation/odata-server>



What's New in OData v4

- Changes
 - Entity Data Model changes
 - Metadata and data type changes
 - URL changes
 - Changed/added operators
 - Semantics
 - Batch handling, verbs usage, upserts, async
 - Formats
 - JSON verbosity levels
- Using
 - Different NuGet package for server side
 - No scaffolding
 - Can scaffold, but then need to change namespaces and a few other things
 - Client code gen through T4 template instead of Add Service Ref



References

- Pluralsight – Building ASP.NET Web API OData Services
<http://www.pluralsight.com/courses/aspnetwebapi-odata>
- OData.org
<http://www.odata.org>
- ASP.NET Web Stack
<http://aspnetwebstack.codeplex.com>
- REST in Practice, Jim Webber et al,
<http://shop.oreilly.com/product/9780596805838.do>

 brian.noyes@solliance.net

 [@briannoyes](https://www.facebook.com/briannoyes)

 <http://briannoyes.net>



DEV  intersection
europe 2015

<anglebrackets/>
europe 2015

Connect with me

Brian Noyes

brian.noyes@solliance.net

[@briannoyes](https://www.facebook.com/briannoyes)

<http://briannoyes.net>

<http://www.solliance.net>

My Connect

On the page for this session, you'll find

- Ask a question to the speaker(s)
- Slides
- Resources
- Feedback form