

Introduction to RESTful Web Services  
David Neal

# REST Easy

# What is REST?

---

- Representational State Transfer
- Not a Standard
- Architectural Style
- Set of Principles or Design Pattern

# Why REST?

- Everybody's Doing It!
- WDNNSA
  - We Don't Need No Stinkin' Abstractions
- Simplicity
- jQuery
- Bacon for the Web

# Evolved RESTful Web Principles

- URL / URI
  - Subjective
  - Simplicity
  - Nouns over verbs
  - “Hackable”
- HTTP Verbs
  - GET, POST, PUT, DELETE, HEAD, OPTIONS

# REST Examples

- GET /orders
- GET /orders/45
- GET /customers/32/orders
- GET /customers/32/orders/15
- POST /orders
- PUT /orders/45
- DELETE /orders/45

# Evolved RESTful Web Principles

- Status
  - 200 – OK
  - 201 – Created
  - 202 – Accepted
  - 401 – Unauthorized
  - 404 – Not Found
- Metadata
- Representations
- Hypertext/Hypermedia
  - Contextual Links

# Consuming REST in .NET

---

- `System.Net.WebRequest`
- `Hammock`
- `RestSharp`

# Awesome REST Client Demos



# Creating RESTful APIs

---

- MVC
- WCF Web API (currently Preview 4)
- OpenRasta
- Node.js

# Even More Awesomer REST Server Demo

# Resources

- O'Reilly "RESTful Web Services" and "RESTful Web Services Cookbook"
- <http://restpatterns.org>
- "Build Truly RESTful API and website using same ASP.NET MVC code" on <http://codeproject.com>
- <https://dev.twitter.com/docs>
- <http://wcf.codeplex.com/>
- Roy Fielding's Dissertation

# Thank You!

David Neal

<http://celljournalist.com>

Email: david@reverentgeek.com

Twitter: @reverentgeek

Blog: <http://reverentgeek.com/>  
(coming soon... er... someday)

Source Code

<https://github.com/reverentgeek/RestSamples>