

Implementing an API in ASP.NET Web API

Module 4: Versioning

Shawn Wildermuth
Wilder Minds LLC
<http://wilder minds.com>



pluralsight 
hardcore developer training

Agenda

- **Versioning**

- API Versioning
- Creating a Versioned Controller
- Using Routing to Version with URLs
- ApiControllerSelector Explained
- Versioning with the Query String
- Versioning with a Version Header
- Versioning with an MediaType

What is API Versioning?

- **Once you publish an API, it's set in stone**
 - Users/Customers rely on the API not changing
 - But requirements will change
 - Need a way to evolve the API without breaking existing clients
 - API Versioning isn't Product Versioning
 - Don't tie them together

The Problem with API Versioning

- **In Typical Projects**

- Versioning is accomplished with different versions of the package
 - E.g. Assemblies

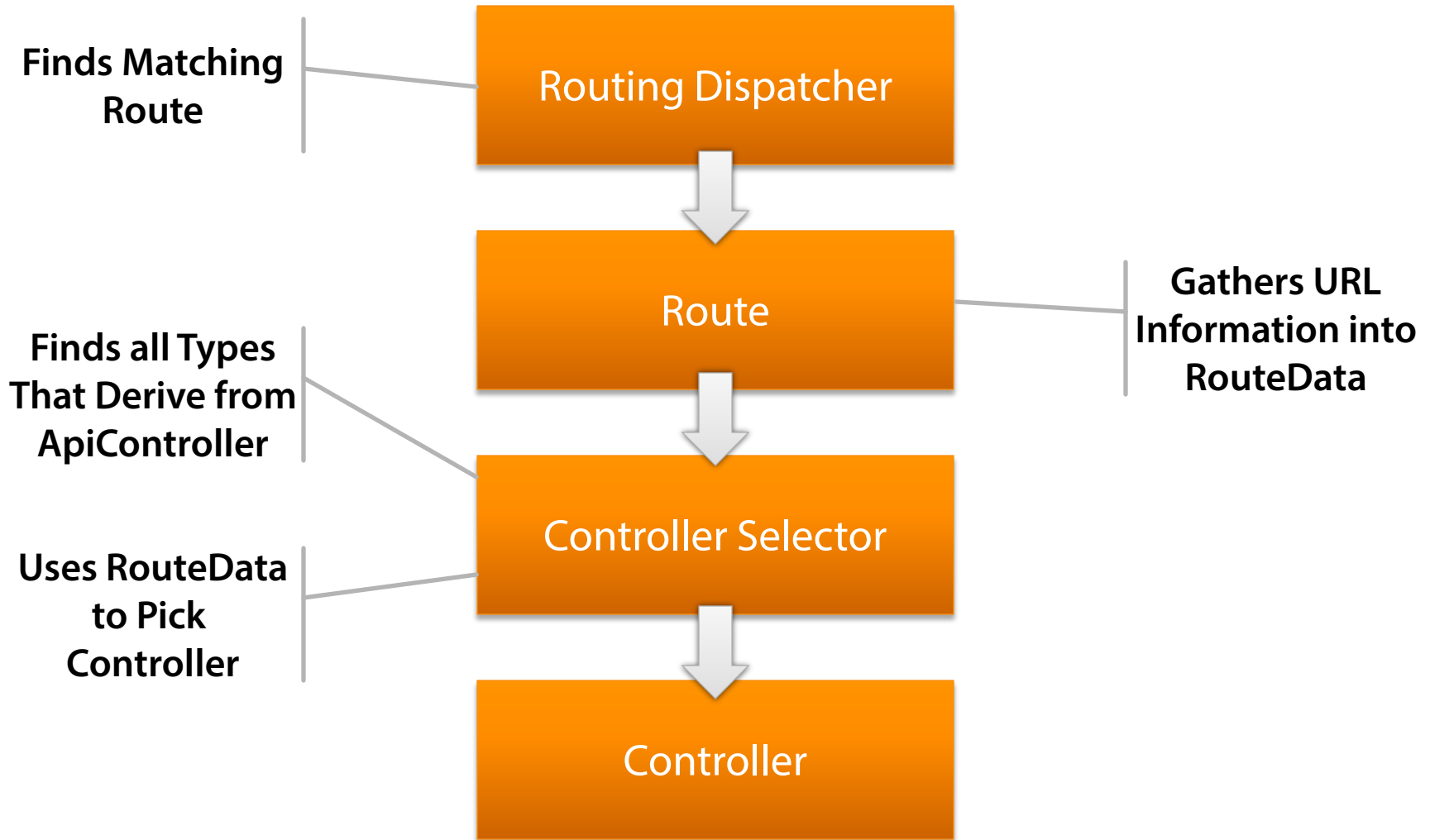
- **API Versioning is Harder**

- Your API needs to support both new and old users
- Side-by-side deployment isn't feasible most of the time
- Need to support both versions in same code base

Creating a Versioned Controller

Using Routing to Version with URLs

How Web API Picks a Controller



Creating a `HttpControllerSelector`

Versioning with the Query String

Versioning with a Version Header

Versioning with Accept Header Version

Versioning with MediaTypes

SDammann.WebApi.Versioning

Summary

■ Versioning

- Versioning is hard but necessary part of publishing an API
- By building versioned controllers, you can enable side-by-side versions
- With the `HttpControllerSelector`, you can direct to versioned controllers
- You can version your API using the URL, Query String or Headers
- Finding the right versioning scheme for your API isn't trivial