infogain

Web-Api development using Asp.Net Core

Coverage

- ASP.NET core, .NET core and .NET framework
- Request Pipeline and Middleware
- Routing
- Status Code
- Content Negotiation
- Formatters
- Resource Manipulation
  - Creating
  - Updating
  - Deleting
- Validating Input
- Inversion of Control and Dependency Injection
  - Logging
  - Custom services

ASP.NET Core

- Framework for building modern internet connected application
- Open-source
  - https://github.com/aspnet/
- Cross-platform
  - Runs on Windows, Mac Linux
  - Develop on Windows, Mac, Linux
- Rethought from the ground up
- Granular set of Nuget packages
- Smaller application surface area
  - Tighter security
  - Reduced Servicing
  - Improved performance

Full .NET Framework

The full framework we know

Target all dependencies we're used to

.NET Core

Modular version of .NET framework

Portable across platform

Subset of .NET framework

Performance Improvements

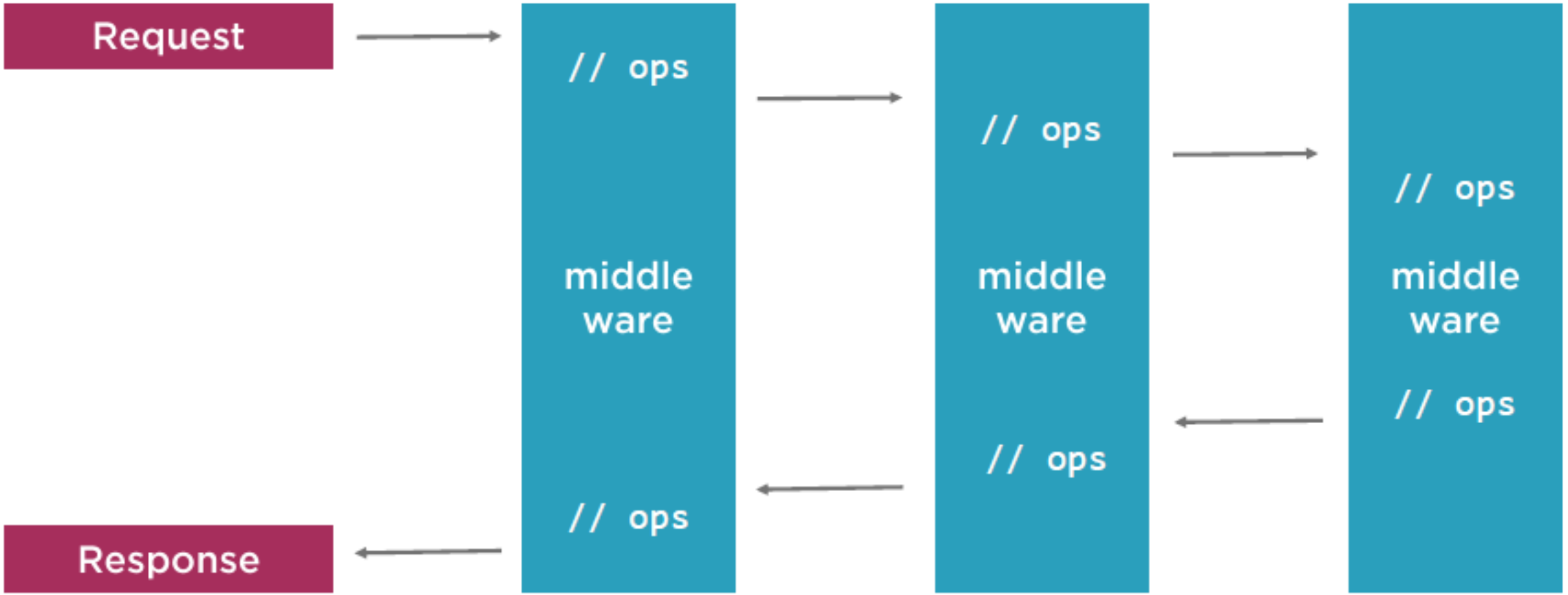
Implementation of .NET standard

- Choosing an environment
  - Licences, functionality or even personal preference

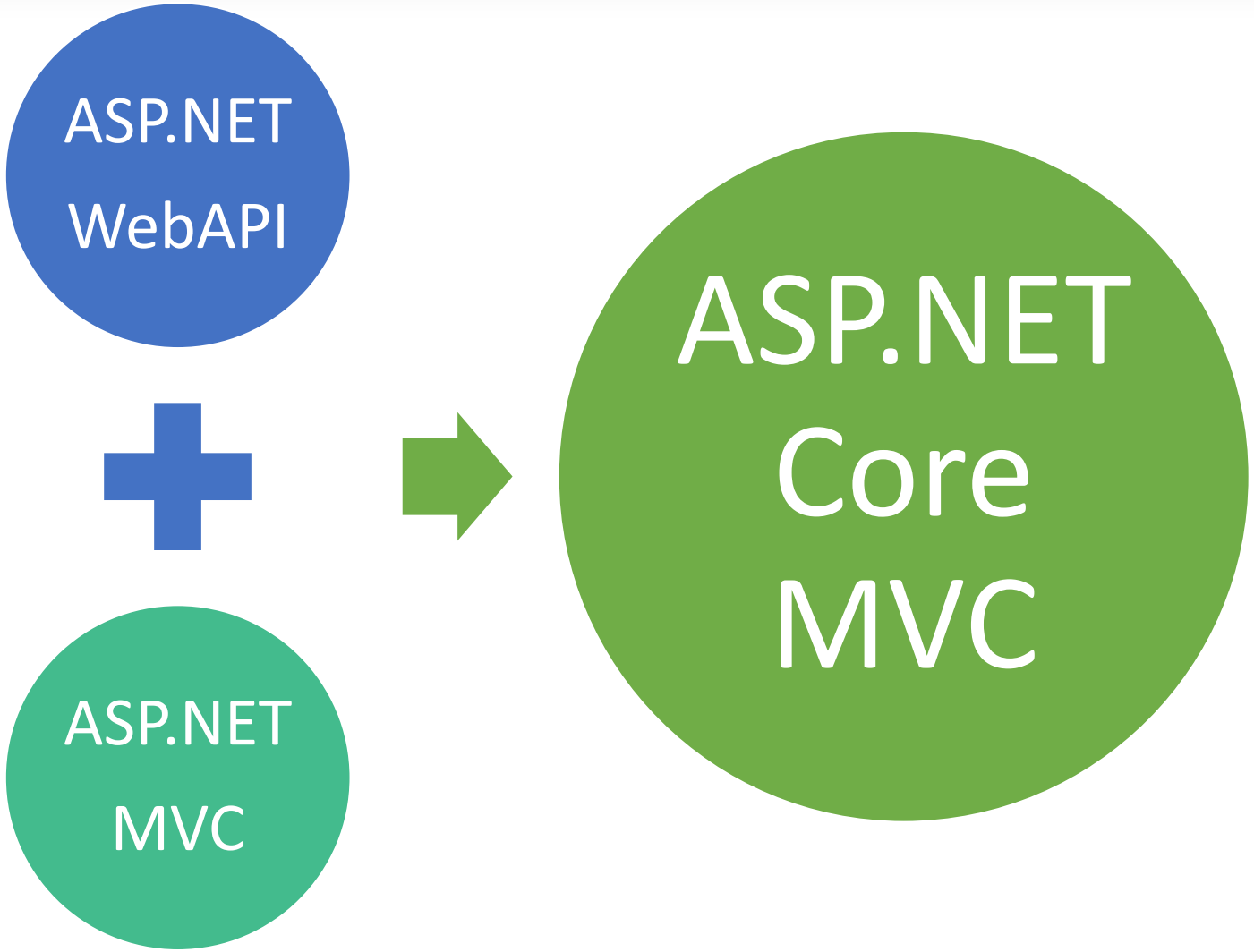
- Choosing between ASP.NET Core 1 & 2
  - Larger API surface might require you to use a newer version
  - Different support lifecycle might require you to use an older version

The ASP.NET Core Request Pipeline & Middleware

infogain
www.infogain.com

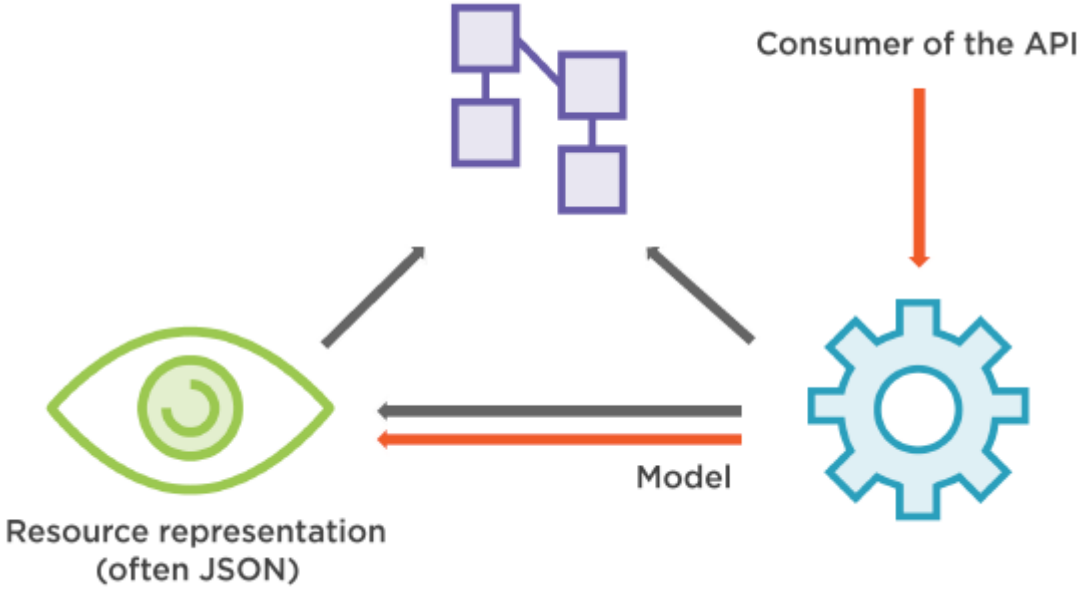


Middleware for Building an API



MVC pattern

- Model – View – Controller
- Architectural pattern
- Loose coupling, separation of concerns : Testability , reuse



REST

- **REST** is the abbreviation of **Representational State Transfer**.
- This term was introduced and defined in 2000 by Roy Fielding in his doctoral dissertation.
- Once a web service adopts REST concepts, we call it a **RESTful Web Service** or **RESTful Web API**
- **REST architectural style Constraints**
  - Client-Server - as a web based system, this is, naturally, a must-have
  - Stateless - **MUST** have
  - Cache - nice to have, especially for large systems
  - Uniform Interface - the central feature **MUST**-have
  - Identification of resources - **MUST**-have
  - Manipulation of resources through representations - **MUST**-have
  - Self-descriptive messages - **MUST**-have
  - Hypermedia as the engine of application state (HATEOAS) - nice to have
  - Layered System - nice to have, especially for large systems
  - Code on demand - optional

ASP.NET Core 2 Metapackage and Runtime Store

- Asp.net core 2.0 introduces the Microsoft.AspNetCore.All
- metapackage
  - Referenced by default for new ASP.NET applications
  - The metapackage adds references to a list of packages

- Microsoft.AspNetCore.All includes :
  - All supported ASP.NET Core packages
  - All supported Entity Framework Core packages
  - Internal and 3rd –party dependencies used by ASP.NET Core and Entity Framework Core

- Microsoft.AspNetCore.App metapackage for ASP.NET Core 2.1

  Includes all supported packages

  except those that contain third-party dependencies

Routing

- Matched request URI to controller method
- Convention-based and attribute-based routing

```
app.UseMvc(config => {
    config.MapRoute(
        name: "Default",
        template: "{controller}/{action}/{id?}",
        defaults: new { controller="Home", action="Index" }
    ); });
```

Convention-based Routing

Conventions need to be configured

Not advised for API's

Attribute-based Routing

Attributes at controller & action level, including an (optional) template

URI is matched to a specific action on a controller

Routing

Http Method	Attribute	Level	Sample URI
Get	HttpGet	Action	/api/restaurants /api/restaurants/1
Post	HttpPost	Action	/api/restaurants
Put	HttpPut	Action	/api/restaurants/1
Patch	HttpPatch	Action	/api/restaurants/1
Delete	HttpDelete	Action	/api/restaurants/1
---	Route	Controller	---

Status Code

infogain
www.infogain.com

Level 200 Success

- 200 - Ok
- 201 - Created
- 204 – No Content

Level 400 Client Error

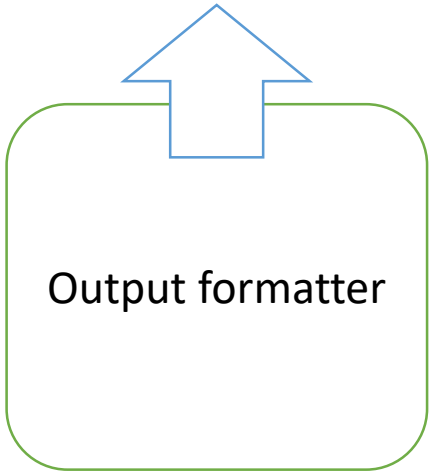
- 400 – Bad Request
- 401 – Unauthorized
- 403- Forbidden
- 404 – Not Found
- 409 - Conflict

Level 500 – server error

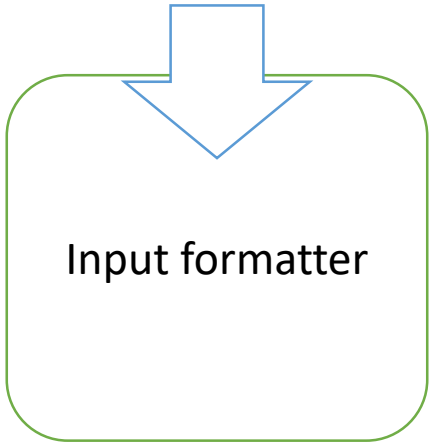
- 500 –Internal Server Error

Formatter and Content Negotiation

- Media type is passes via the Accept header of the request
  - application/json
  - application/xml



Deals with output
Media type : accept header



Deals with input
Media type : content-type header

Day 2

Partially Updating a Resource

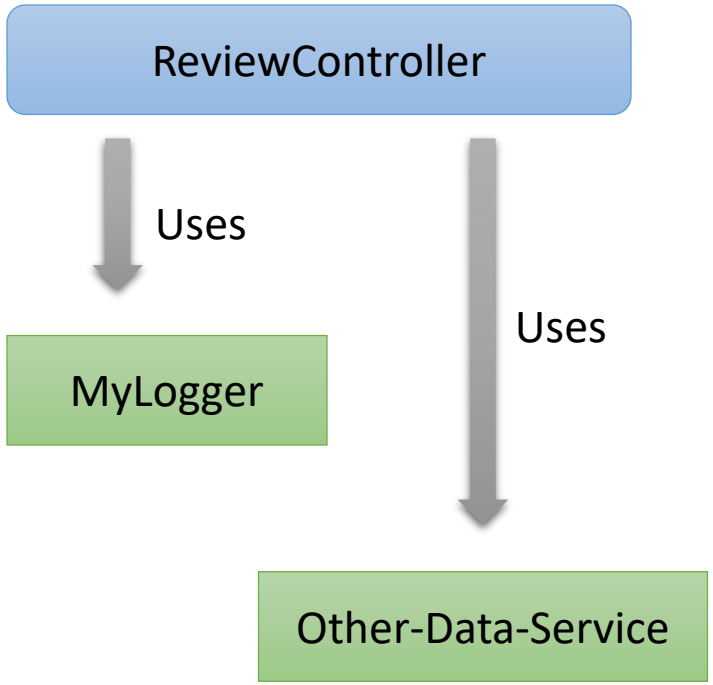
- JsonPatch (RFC 6902)
  https://tools.ietf.org/html/rfc6902
- Describes a document structure for expressing a sequence of operation to apply to a JSON document

```
[
 { "op": "replace", "path": "/comment", "value": "  best parking" },
 { "op": "add", "path": "/rating", "value": 5 }
]
```

- Array of operations
- “replace ” operation
- “comment “property gets value “best parking”

- “replace ” operation
- “rating “property gets value “4”

Inversion of Control and Dependency Injection



Class implementation has to change when a dependency changes

Difficult to test – hard apply the mock version of dependency

Class manages the lifetime of the dependency

This is tight coupling

Inversion of Control

Inversion of Control delegates the function of selecting a concrete implementation type for a class's dependencies to an external component .

Dependency Injection

Dependency Injection is a specialization of the Inversion of Control pattern. The Dependency Injection pattern uses an object – the container - to initialize objects and provide the required dependencies to the object .

Inversion of Control and Dependency Injection

- Dependency Injection is built into ASP.NET Core
- ConfigureServices is used to register services with the built-in container
- Demo – Injecting and Using a Logger

infogain

www.infogain.com

Infogain Corporation, HQ
485 Alberto Way Los Gatos,
CA 95032 USA
Phone: 408-355-6000
Fax: 408-355-7000

Pune
7th Floor, Bhalerao Towers, CTS No.1669 - 1670, Behind Hotel Pride,
Shivaji Nagar, Pune - 411005
Phone : +91-20-66236700

Infogain Irvine
41 Corporate Park,
Suite 390 Irvine, CA 2606 USA
Phone: 949-223-5100
Fax: 949-223-5110

Infogain Austin
Stratum Executive Center Building D
11044 Research Boulevard Suite 200
Austin, Texas 78759

Noida
A-16, Sector 60, Noida Gautam Budh agar,
201301 (U.P.) India
Phone: +91-120-2445144
Fax: +91-120-2580406

Dubai
P O Box 500588 Office No.105,
Building No. 4, Dubai Outsource Zone,
Dubai, United Arab Emirates
Tel: +971-4-458-7336