

# OpenAPI.jl

## For Easy HTTP APIs

JuliaCon 2024

Slides: <https://bit.ly/3wLMRUJ>



Tanmay | JuliaHub Inc. | @tanmaykm | [tanmaykm@gmail.com](mailto:tanmaykm@gmail.com) | [tanmay@juliahub.com](mailto:tanmay@juliahub.com)

---

01

## About OpenAPI

Goals.  
Where does it fit in.

02

## OpenAPI.jl

Overview. What's there  
and what's where.

03

## Dive In

Use OpenAPI Services.  
Create OpenAPI Services.

04

## Summarize, Q&A

---

# 01 About OpenAPI

Goals. Where does it fit in.

{ REST }

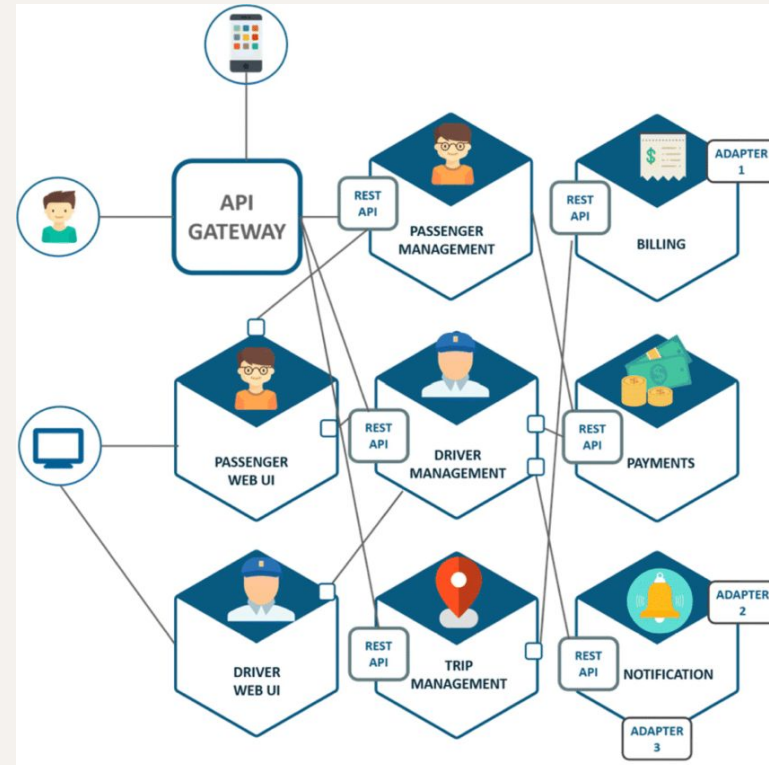
GRPC



# OpenAPI - the specification

- Interface Specification for HTTP APIs
- Leverages REST
  - A service has Resources
  - Resources have States
  - Actions create resources or change states
- Programming language agnostic

Goal: Discover and understand capabilities of a service (server) without access to source code, documentation, or network traffic inspection.



Source:

[https://www.researchgate.net/publication/342124274\\_Plug-and-play\\_mix-and-match\\_a\\_capital\\_systems\\_theory\\_of\\_digital\\_technology\\_platforms](https://www.researchgate.net/publication/342124274_Plug-and-play_mix-and-match_a_capital_systems_theory_of_digital_technology_platforms)

# OpenAPI - tools

## Origins:

- Donated to the OpenAPI Initiative by SmartBear Software.
- Renamed Swagger Specification to OpenAPI specification (2015)
- First official release (v3.0.0) in 2017

## OpenAPI ↔ Swagger

- OpenAPI = Specification (<https://spec.openapis.org/oas/v3.1.0>).
- Swagger = Tools for implementing the specification (<https://swagger.io/tools/>)

## Other Tools

- The code generator (e.g. openapi-generator, kiota, ...)
- Quite a lot of third party tools (e.g. Redocly, ...)

# OpenAPI

- Specification for REST APIs
- Tools to
  - Create Specifications
  - Generate Code, Documents, Tests

Simple and widely understood.

More general purpose than gRPC or GraphQL.

Lightweight compared to SOAP.



# OpenAPI - specification

- Servers
- Tags
- Paths
  - Methods
    - Parameters
    - Responses
- Components

Example:

[https://raw.githubusercontent.com/JuliaComputing/OpenAPI.jl/main/test/specs/petstore\\_v3.json](https://raw.githubusercontent.com/JuliaComputing/OpenAPI.jl/main/test/specs/petstore_v3.json)

# 02

## OpenAPI.jl

What's there. And where.





# Specification Implementation

## OpenAPI Code Generator

<https://openapi-generator.tech/>

## OpenAPI.jl

<https://github.com/JuliaComputing/OpenAPI.jl>

## Swagger.jl

<https://github.com/JuliaComputing/Swagger.jl>  
soon to be deprecated

# API Integrations

## Kubernetes

Kuber.jl

## Azure Cloud

Azure.jl

## Collaboration Tools

E.g. GitHub, Slack, ...

---

# Code Generator

**Implementation** (<https://github.com/OpenAPITools/openapi-generator>)

```
modules/openapi-generator/src/main  
  resources/julia-  
    java/org/openapitools/codegen/languages/*Julia*
```

## Download and Install (requires Java)

Instructions at <https://github.com/OpenAPITools/openapi-generator>

## Use Tools from OpenAPI.jl

Local Generator Service (requires docker)

Remote Generator Service

One hosted in public internet by OpenAPI org

---

---

# Generated Code Organization

- Generated as Julia module
- Docs generated in separate folder
- Structured similar to a Julia package
  
- Client modules are complete
  - Based on Downloads.jl
- Server modules need implementation of server methods
  - Based on HTTP.jl

# 03

## Dive In



---

# Demo!

- Look at a specification
  - Generate code (client)
  - Look at the generated code
  - Use the client
  - Generate both client and server code
  - Implement server methods
  - Run the server. Use it.
-

# 04

## Summarize, Q&A



---

# Summarizing...

- Discussed OpenAPI and related standards
- OpenAPI.jl package and where it stands
  
- Used existing service (GitHub) as a client
- Defined a specification
- Implemented and Deployed a Service

# Q&A

Slides:

<https://bit.ly/3wLMRUJ>



Tanmay | JuliaHub Inc. | @tanmaykm | tanmaykm@gmail.com | tanmay@juliahub.com



# Thanks

Tanmay | JuliaHub Inc. | @tanmaykm | tanmaykm@gmail.com | tanmay@juliahub.com

**CREDITS:** This presentation template was created by **Slidesgo**, including icons by **Flaticon**, infographics & images by **Freepik**