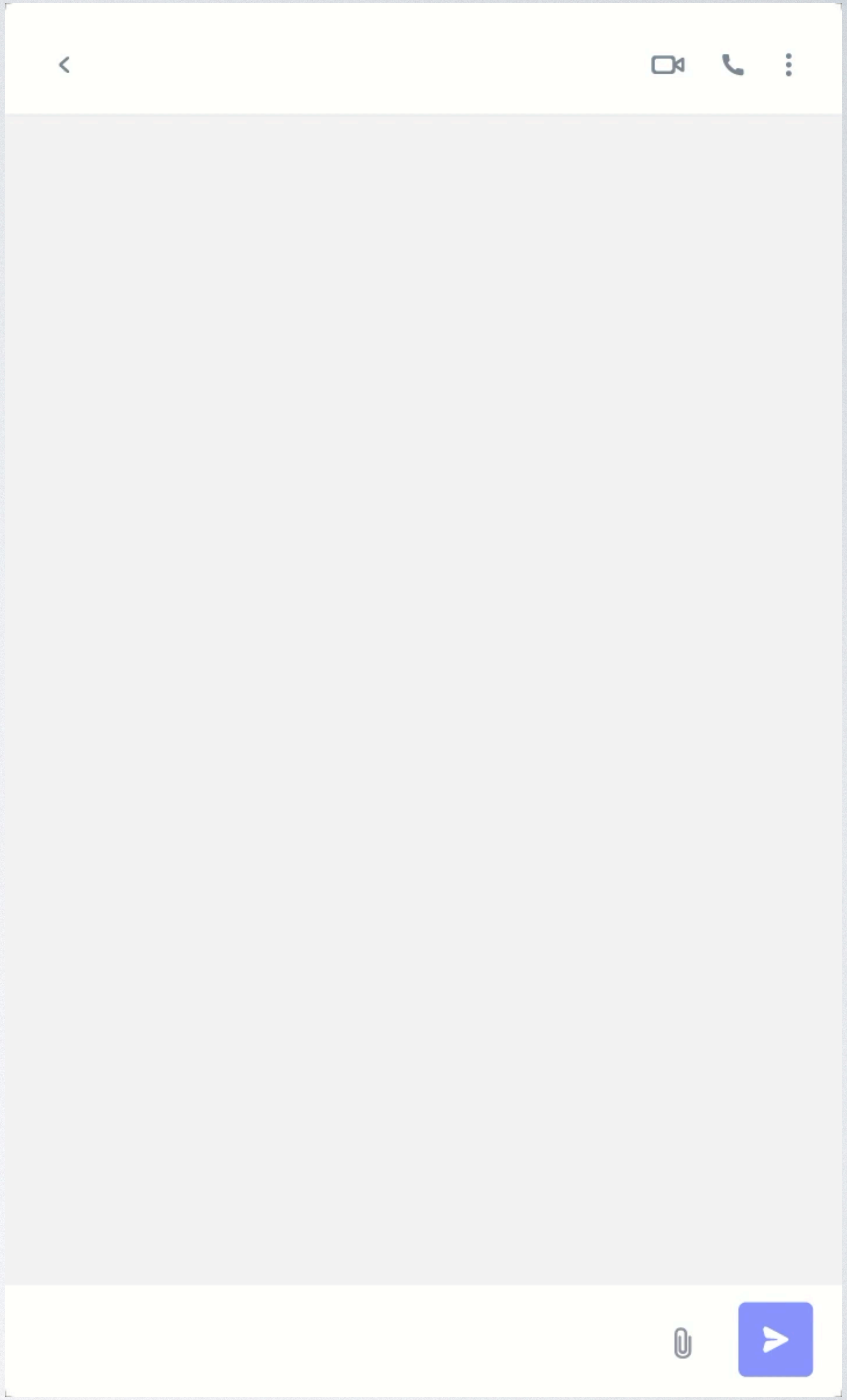
Utiliser OpenAPI plutôt qu’un document HTML quelconque pour décrire son API

Voxxed Days Luxembourg 2019 - Jérémie Bresson unblu

Jérémie Bresson

@j2r2b

jmini



Visitor
V
End Co-Browsing
End Conversation
Homepage - Voxxed Days Luxembourg
https://voxxeddays.com/luxembourg/
HOME
CALL FOR PAPER
REGISTRATION
SPEAKERS
SCHEDULE
MEET & GREET
CONTACT
VOXXED DAYS
LUXEMBOURG
ADD TO MY CALENDAR

20 et 21 juin 2019

Casino 2000, Luxembourg

Voxxed Days Luxembourg est un évènement IT destiné aux développeurs, issu de la famille Devoxx.

Les 3 premières éditions de Voxxed Days ont eu lieu à Luxembourg, les 22 juin 2016, 2017 et 2018! Ces journées étaient composées de conférences sur les technologies futures, pour les développeurs, et par les développeurs.

Cette année nous verrons la 4ème édition!

20 et 21 juin 2019

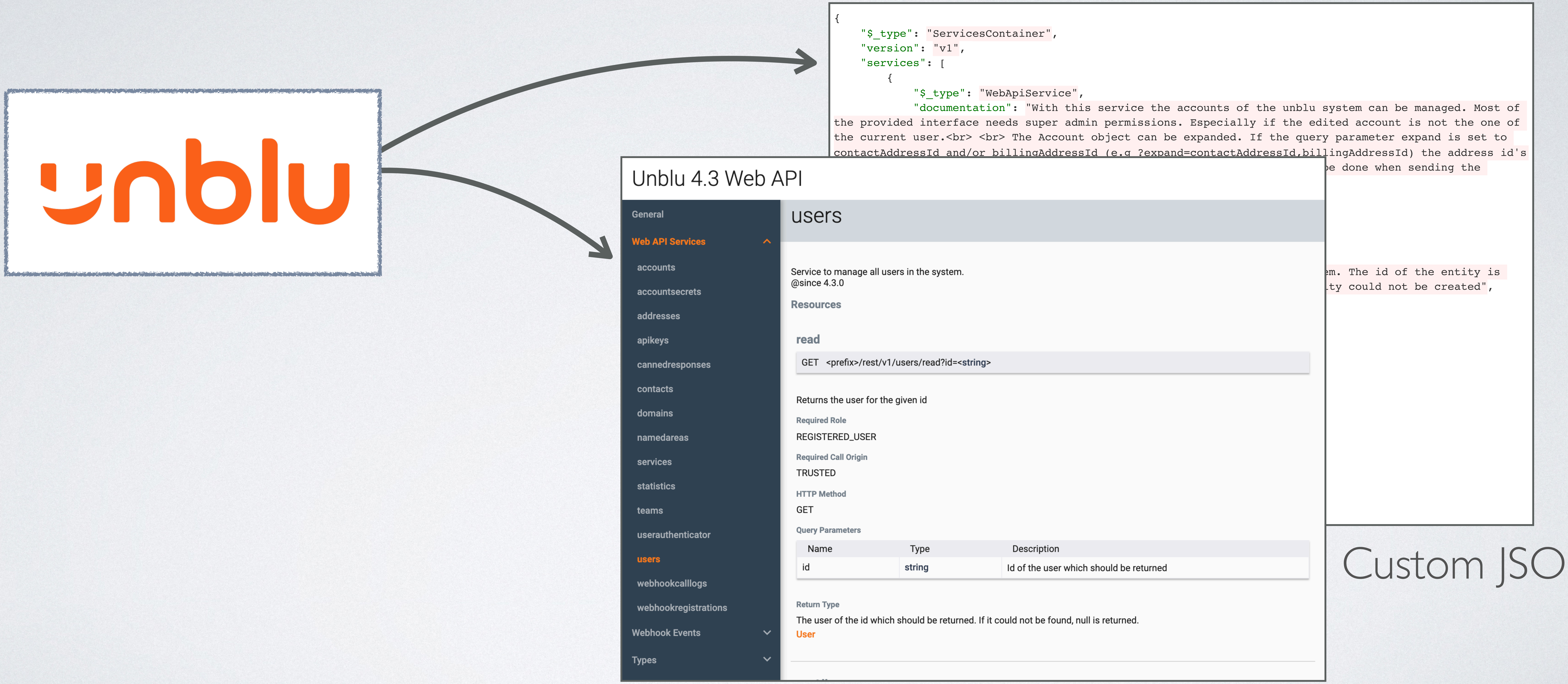
Casino 2000, Luxembourg

Voxxed Days Luxembourg est un évènement IT destiné aux développeurs, issu de la famille Devoxx.

Les 3 premières éditions de Voxxed Days ont eu lieu à Luxembourg, les 22 juin 2016, 2017 et 2018! Ces journées étaient composées de conférences sur les technologies futures, pour les développeurs, et par les développeurs.

Cette année nous verrons la 4ème édition!

At the beginning…



```
{
    "$_type": "ServicesContainer",
    "version": "v1",
    "services": [
        {
            "$_type": "WebApiService",
            "documentation": "With this service the accounts of the unblu system can be managed. Most of
the provided interface needs super admin permissions. Especially if the edited account is not the one of
the current user.<br> <br> The Account object can be expanded. If the query parameter expand is set to
contactAddressId and/or billingAddressId (e.g ?expand=contactAddressId,billingAddressId) the address id's
                                                                                       be done when sending the

                                                                                       m. The id of the entity is
                                                                                       ty could not be created",
```

Unblu 4.3 Web API

users

Service to manage all users in the system.
@since 4.3.0

Resources

read

GET <prefix>/rest/v1/users/read?id=<string>

Returns the user for the given id

Required Role
REGISTERED_USER

Required Call Origin
TRUSTED

HTTP Method
GET

Query Parameters

Name	Type	Description
id	string	Id of the user which should be returned

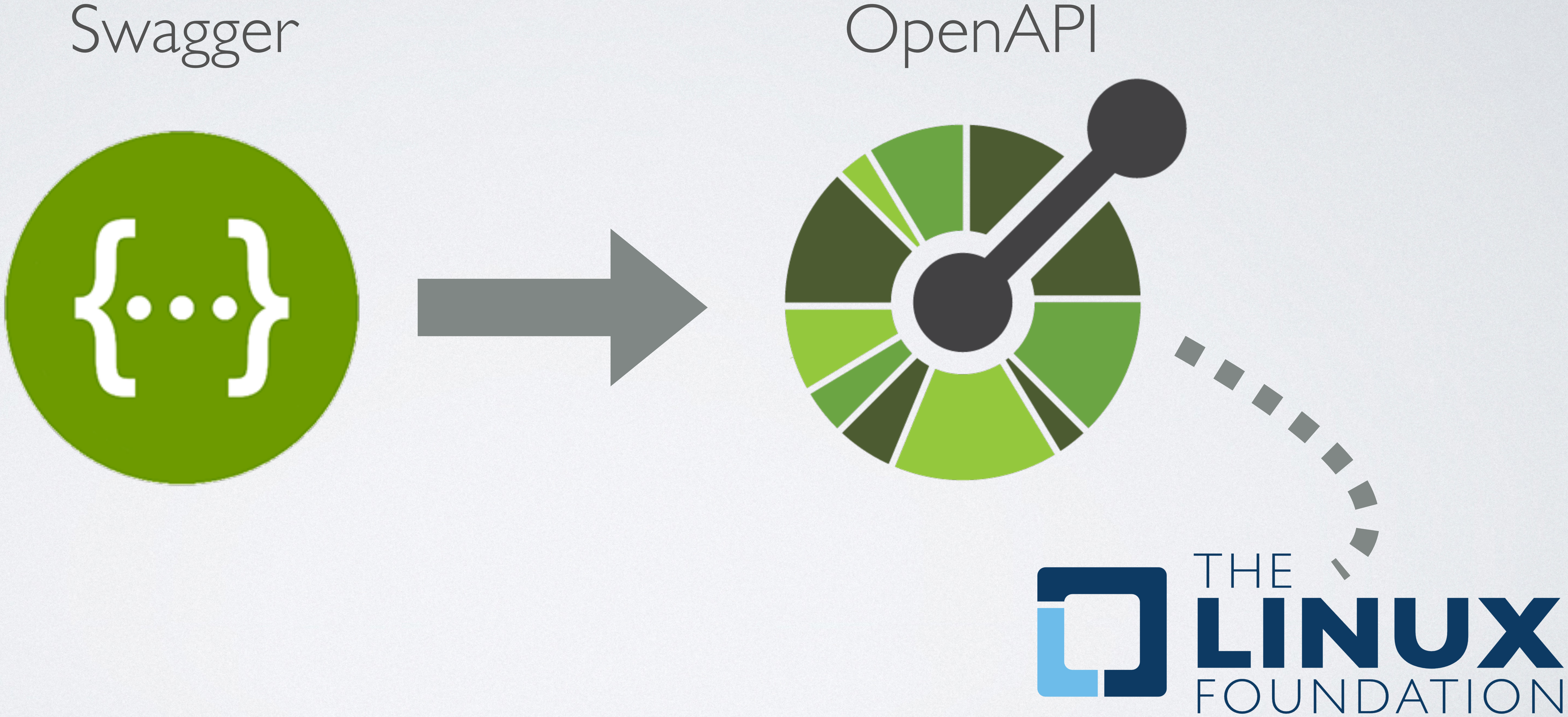
Return Type
The user of the id which should be returned. If it could not be found, null is returned.
User

Custom JSON

Static HTML pages

OpenAPI

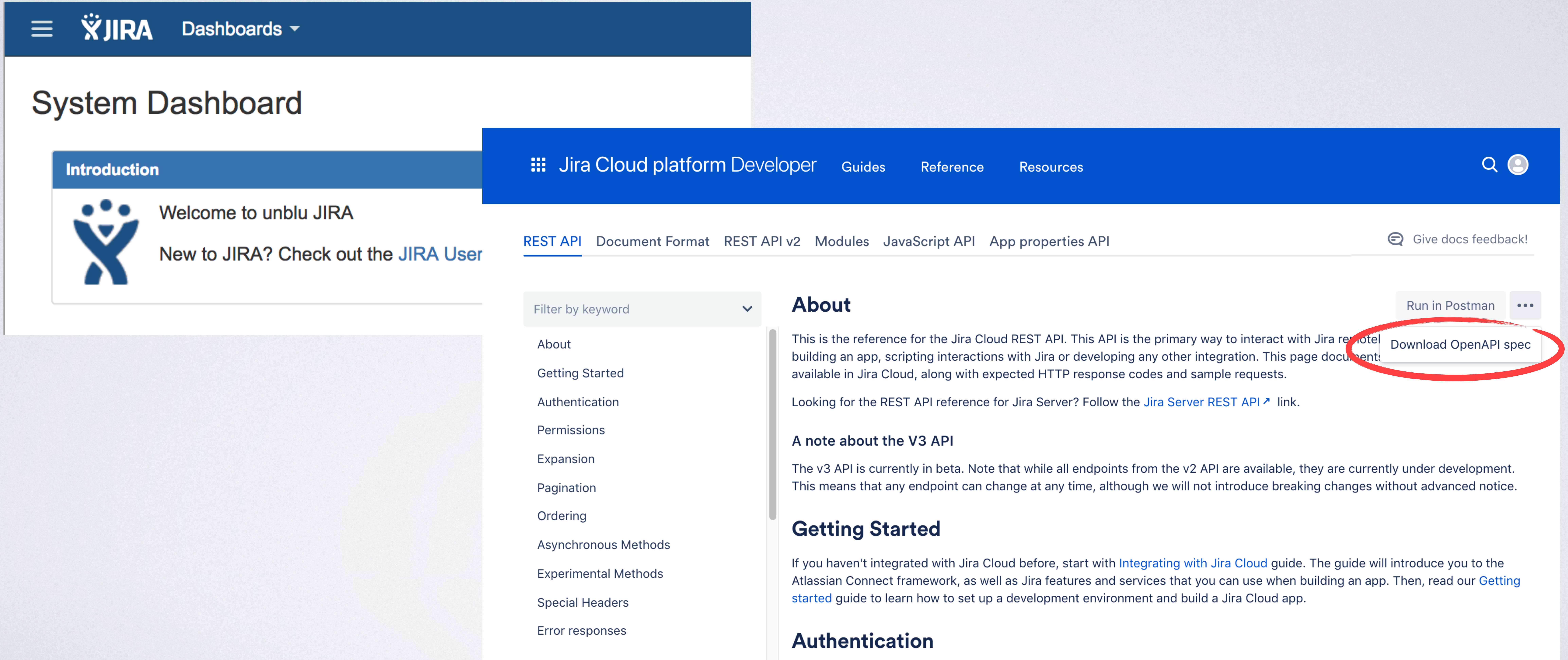
All the same



Swagger

OpenAPI

OpenAPIs are everywhere



System Dashboard

Introduction

Welcome to unblu JIRA

New to JIRA? Check out the JIRA User

About

This is the reference for the Jira Cloud REST API. This API is the primary way to interact with Jira remotel building an app, scripting interactions with Jira or developing any other integration. This page documents available in Jira Cloud, along with expected HTTP response codes and sample requests.

Looking for the REST API reference for Jira Server? Follow the Jira Server REST API ↗ link.

A note about the V3 API

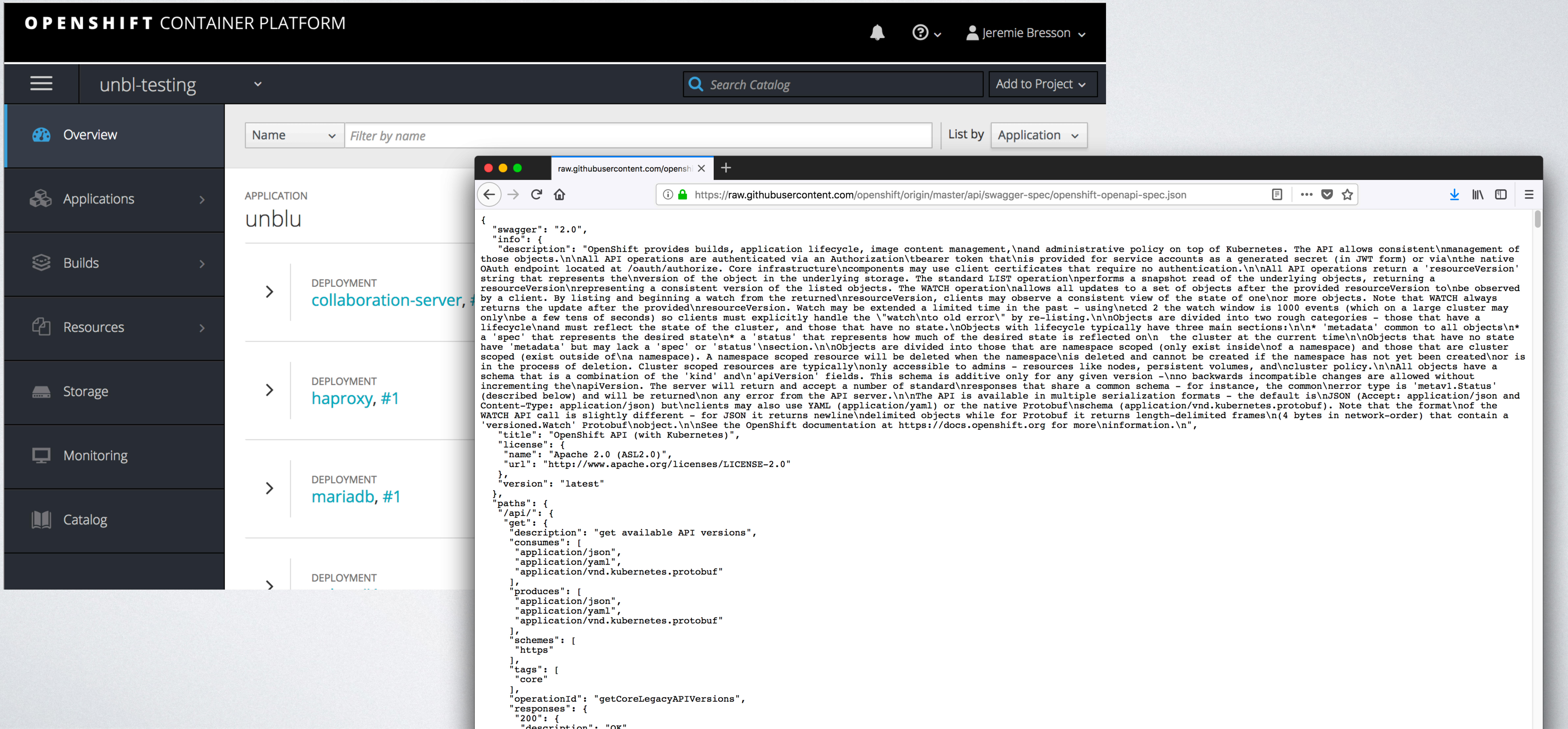
The v3 API is currently in beta. Note that while all endpoints from the v2 API are available, they are currently under development. This means that any endpoint can change at any time, although we will not introduce breaking changes without advanced notice.

Getting Started

If you haven't integrated with Jira Cloud before, start with Integrating with Jira Cloud guide. The guide will introduce you to the Atlassian Connect framework, as well as Jira features and services that you can use when building an app. Then, read our Getting started guide to learn how to set up a development environment and build a Jira Cloud app.

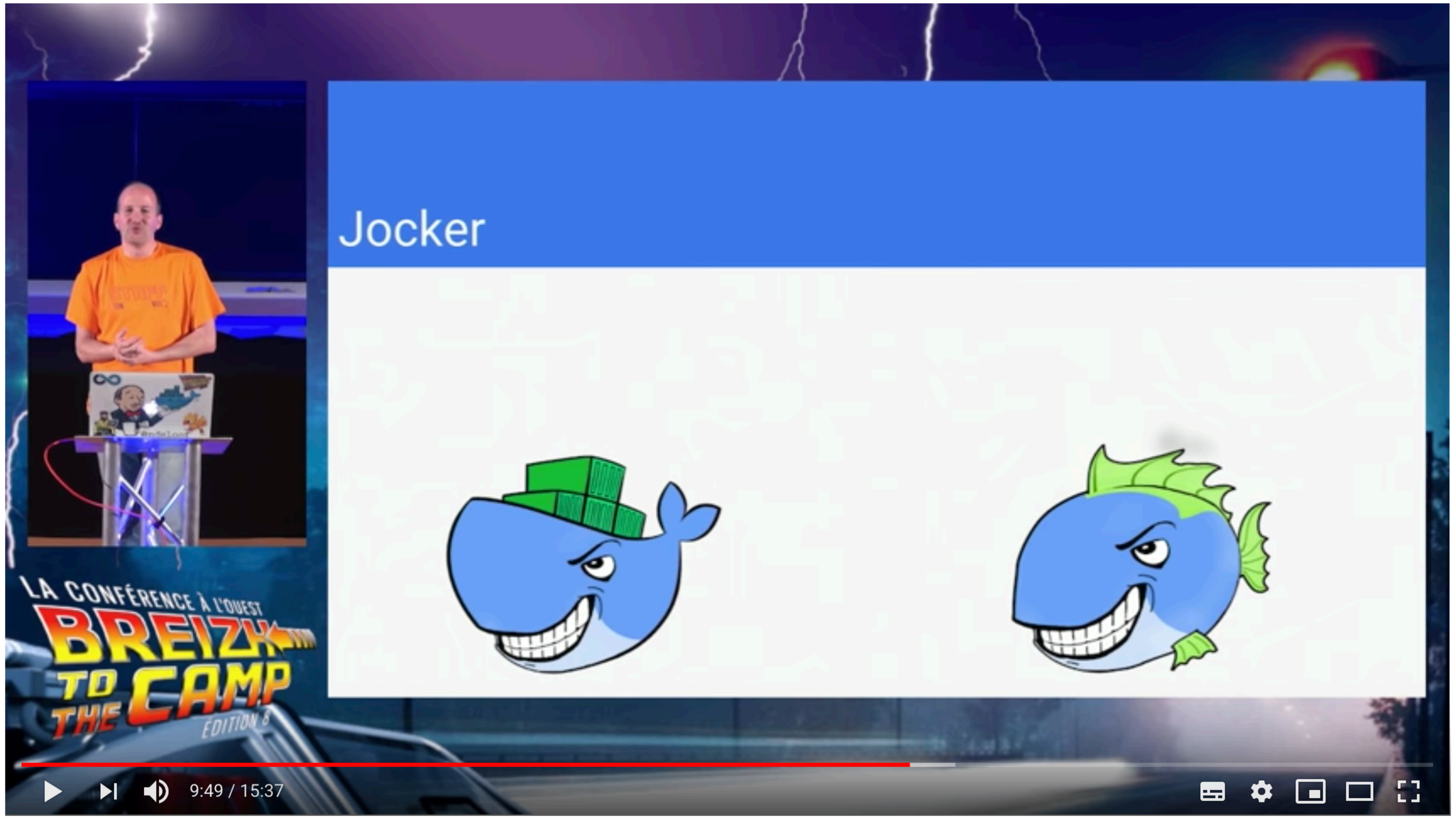
Authentication

OpenAPIs are everywhere



```
{
  "swagger": "2.0",
  "info": {
   "description": "OpenShift provides builds, application lifecycle, image content management,\nand administrative policy on top of Kubernetes. The API allows consistent\nmanagement of
those objects.\n\nAll API operations are authenticated via an Authorization\tbearer token that\nis provided for service accounts as a generated secret (in JWT form) or via\nthe native
OAuth endpoint located at /oauth/authorize. Core infrastructure\ncomponents may use client certificates that require no authentication.\n\nAll API operations return a 'resourceVersion'
string that represents the\nversion of the object in the underlying storage. The standard LIST operation\nperforms a snapshot read of the underlying objects, returning a
resourceVersion\nrepresenting a consistent version of the listed objects. The WATCH operation\nallows all updates to a set of objects after the provided resourceVersion to\nbe observed
by a client. By listing and beginning a watch from the returned\nresourceVersion, clients may observe a consistent view of the state of one\nor more objects. Note that WATCH always
returns the update after the provided\nresourceVersion. Watch may be extended a limited time in the past - using\netcd 2 the watch window is 1000 events (which on a large cluster may
only\nbe a few tens of seconds) so clients must explicitly handle the \"watch\nto old error\" by re-listing.\n\nObjects are divided into two rough categories - those that have a
lifecycle\nand must reflect the state of the cluster, and those that have no state.\nObjects with lifecycle typically have three main sections:\n\n* 'metadata' common to all objects\n*
a 'spec' that represents the desired state\n* a 'status' that represents how much of the desired state is reflected on\n  the cluster at the current time\n\nObjects that have no state
have 'metadata' but may lack a 'spec' or 'status'\nsection.\n\nObjects are divided into those that are namespace scoped (only exist inside\nof a namespace) and those that are cluster
scoped (exist outside of\na namespace). A namespace scoped resource will be deleted when the namespace\nis deleted and cannot be created if the namespace has not yet been created\nor is
in the process of deletion. Cluster scoped resources are typically\nonly accessible to admins - resources like nodes, persistent volumes, and\ncluster policy.\n\nAll objects have a
schema that is a combination of the 'kind' and\n'apiVersion' fields. This schema is additive only for any given version -\nno backwards incompatible changes are allowed without
incrementing the\napiVersion. The server will return and accept a number of standard\nresponses that share a common schema - for instance, the common\nerror type is 'metav1.Status'
(described below) and will be returned\non any error from the API server.\n\nThe API is available in multiple serialization formats - the default is\nJSON (Accept: application/json and
Content-Type: application/json) but\nclients may also use YAML (application/yaml) or the native Protobuf\nschema (application/vnd.kubernetes.protobuf). Note that the format\nof the
WATCH API call is slightly different - for JSON it returns newline\ndelimited objects while for Protobuf it returns length-delimited frames\n(4 bytes in network-order) that contain a
'versioned.Watch' Protobuf\nobject.\n\nSee the OpenShift documentation at https://docs.openshift.org for more\ninformation.\n",
   "title": "OpenShift API (with Kubernetes)",
   "license": {
    "name": "Apache 2.0 (ASL2.0)",
    "url": "http://www.apache.org/licenses/LICENSE-2.0"
   },
   "version": "latest"
  },
  "paths": {
   "/api/": {
    "get": {
     "description": "get available API versions",
     "consumes": [
      "application/json",
      "application/yaml",
      "application/vnd.kubernetes.protobuf"
     ],
     "produces": [
      "application/json",
      "application/yaml",
      "application/vnd.kubernetes.protobuf"
     ],
     "schemes": [
      "https"
     ],
     "tags": [
      "core"
     ],
     "operationId": "getCoreLegacyAPIVersions",
     "responses": {
      "200": {
       "description": "OK"
```

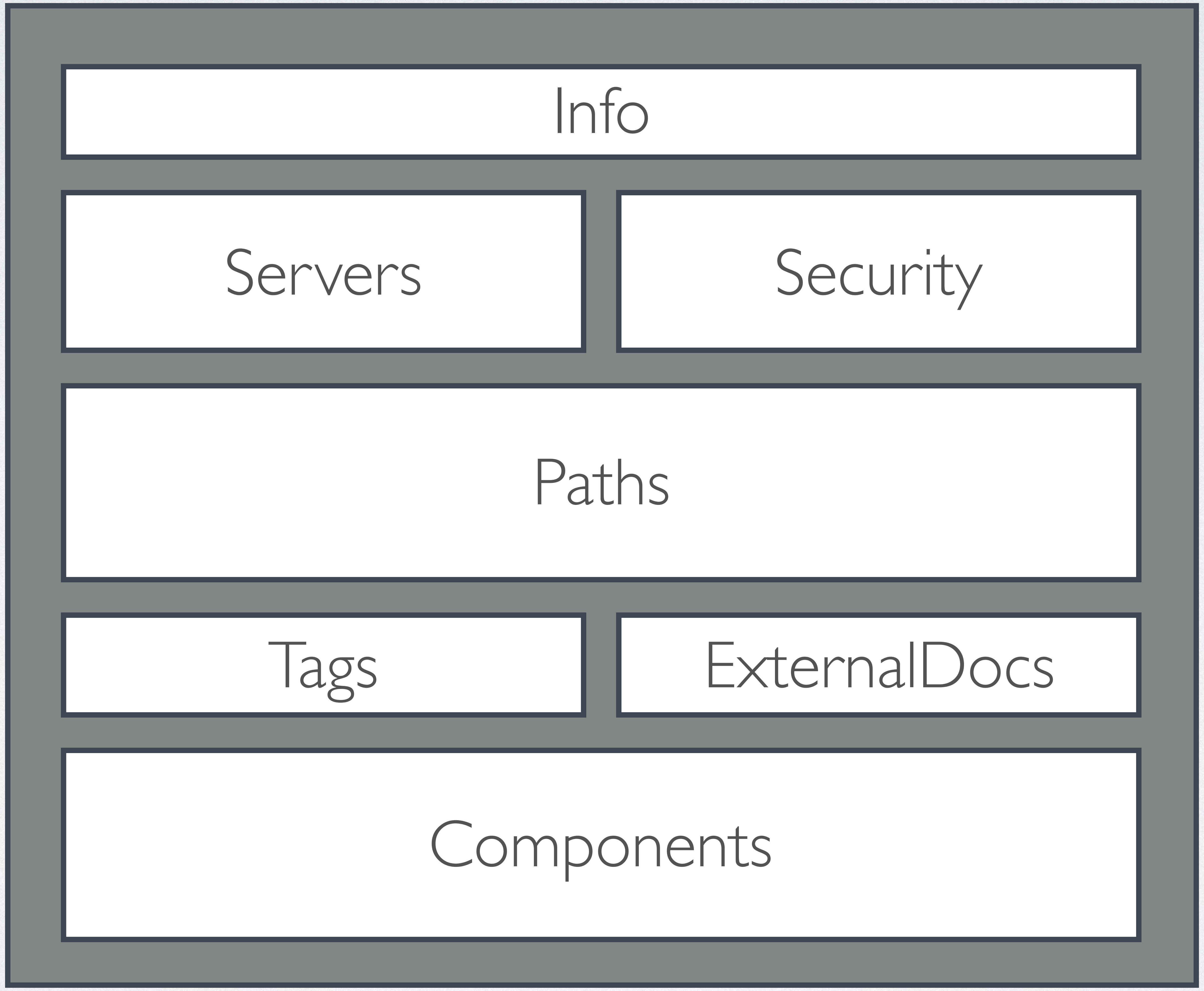
OpenAPIs are everywhere



Jocker, a Java Docker client lib. Où "comment les devs Java aiment se faire mal à... (Nicolas De Loof)

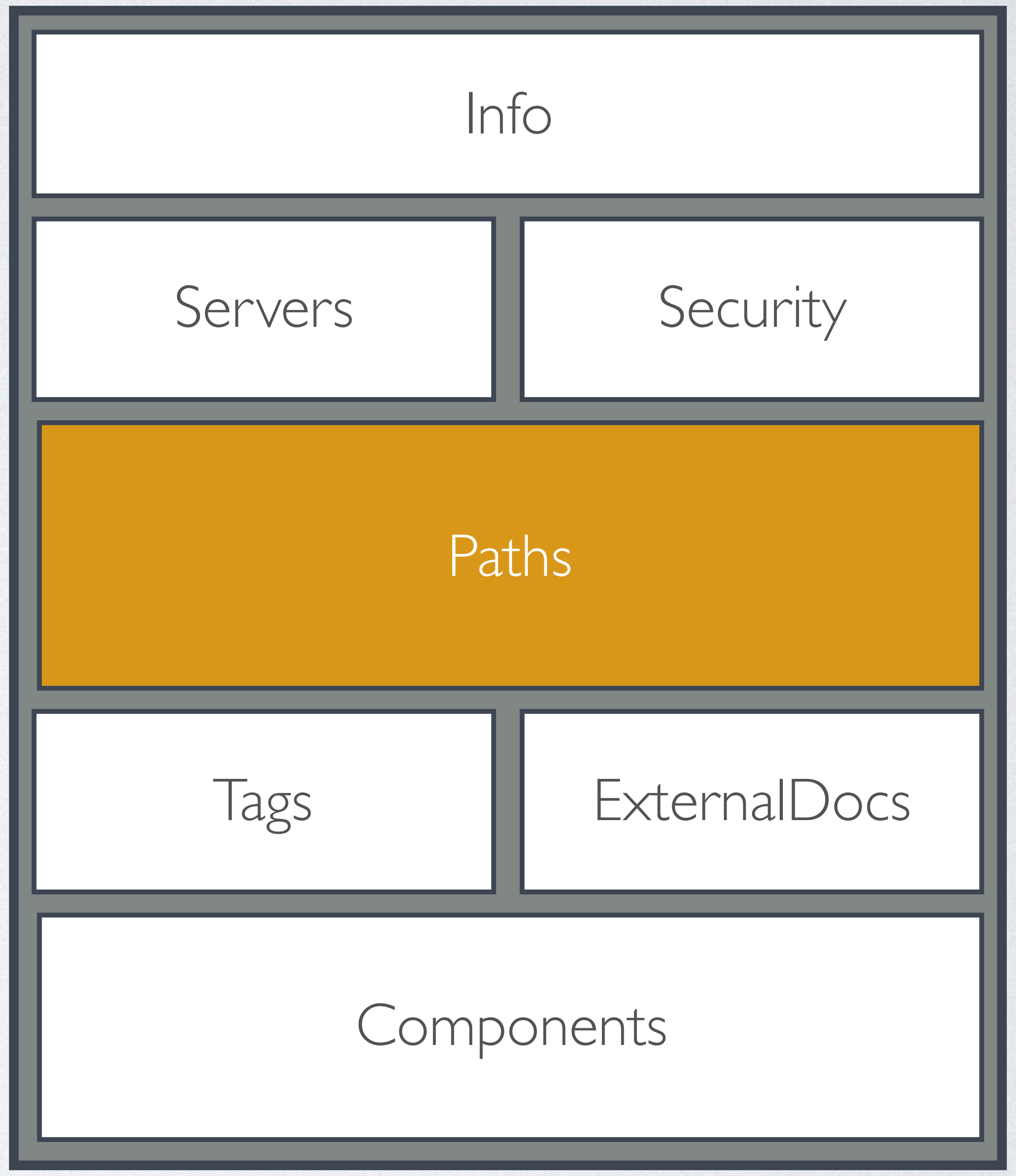
An OpenAPI Specification

OpenAPI v3



JSON
or
YAML

```yaml
openapi: 3.0.1
info:
  title: Todo Backend
  version: "1.0"
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```



Info

Servers

Security

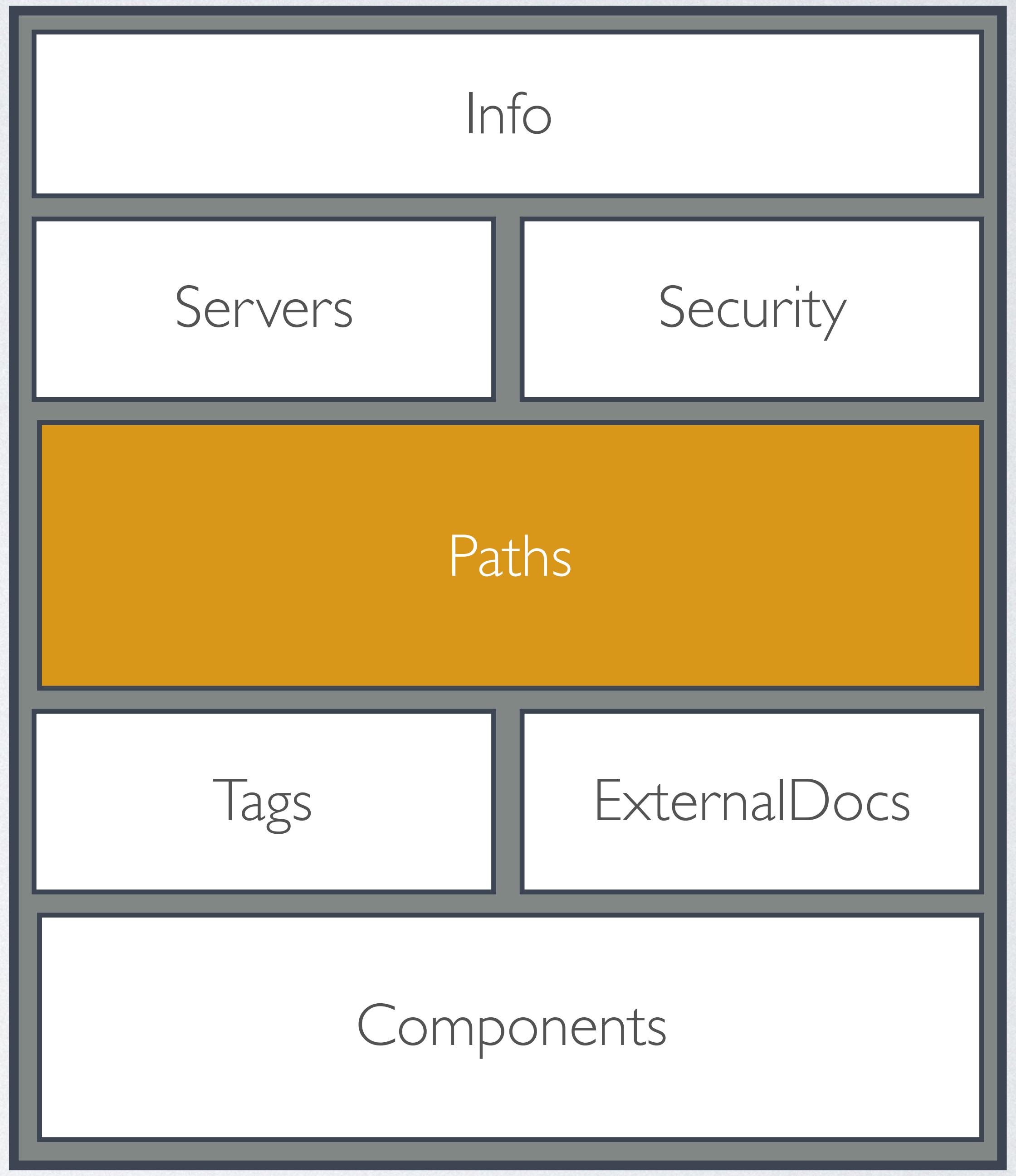
Paths

Tags

ExternalDocs

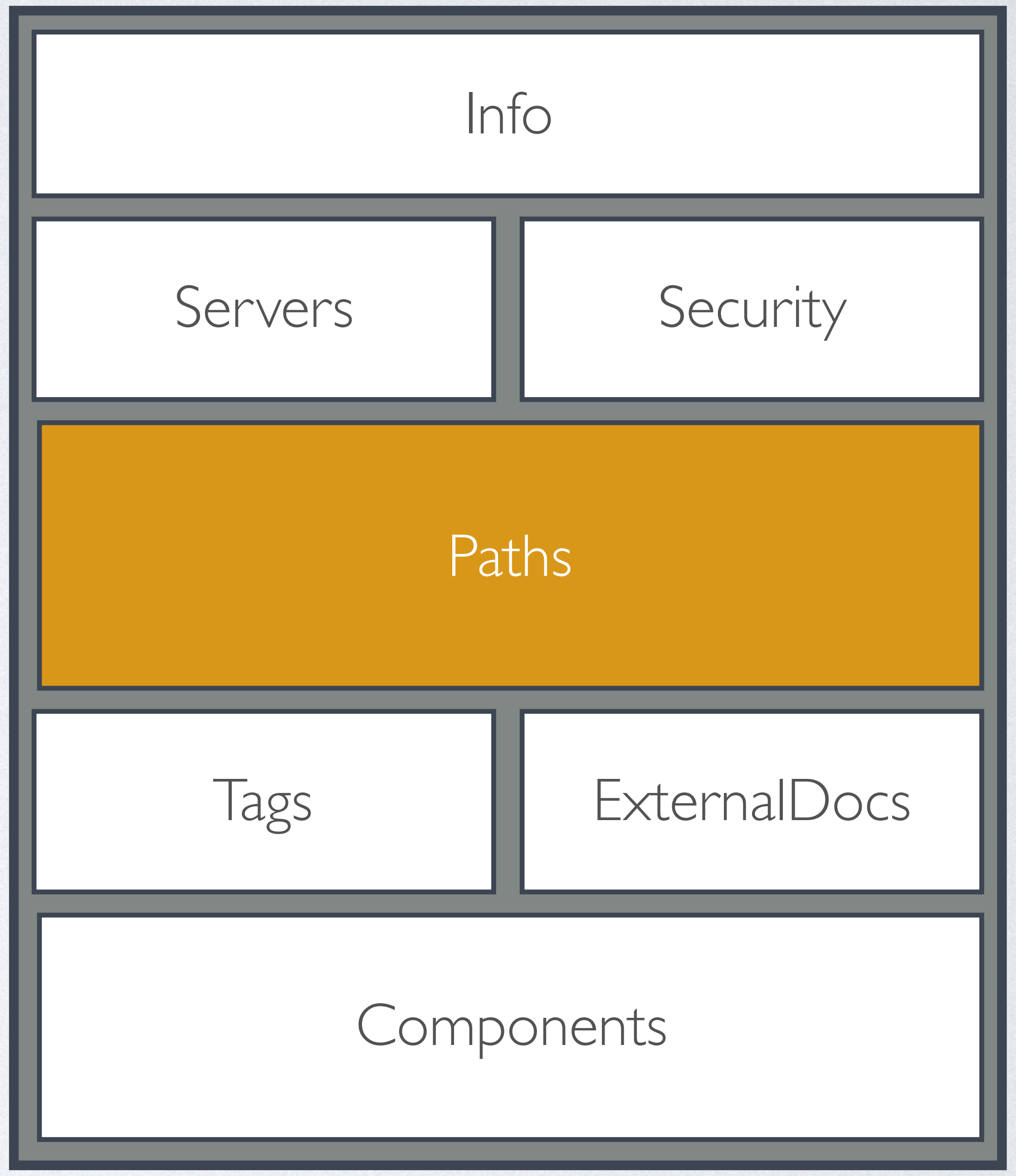
Components

```yaml
openapi: 3.0.1
info:
  title: Todo Backend
  version: "1.0"
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```



```
openapi: 3.0.1
info:
  title: Todo Backend
  version
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```

get, post, patch, delete, put, options, head

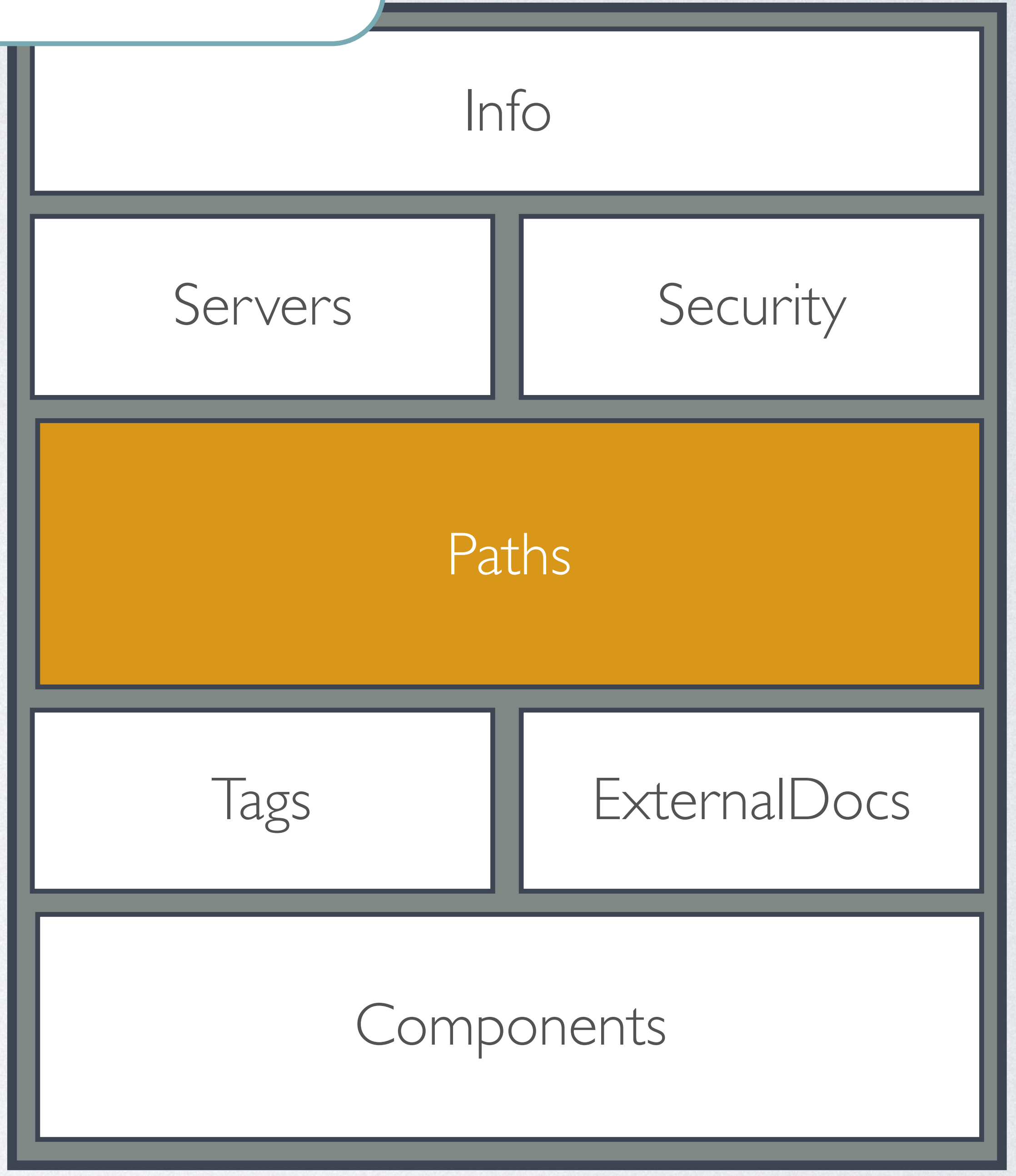


```
openapi: 3.0.1
info:
  title: Todo Backend
  version: "1.0"
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```

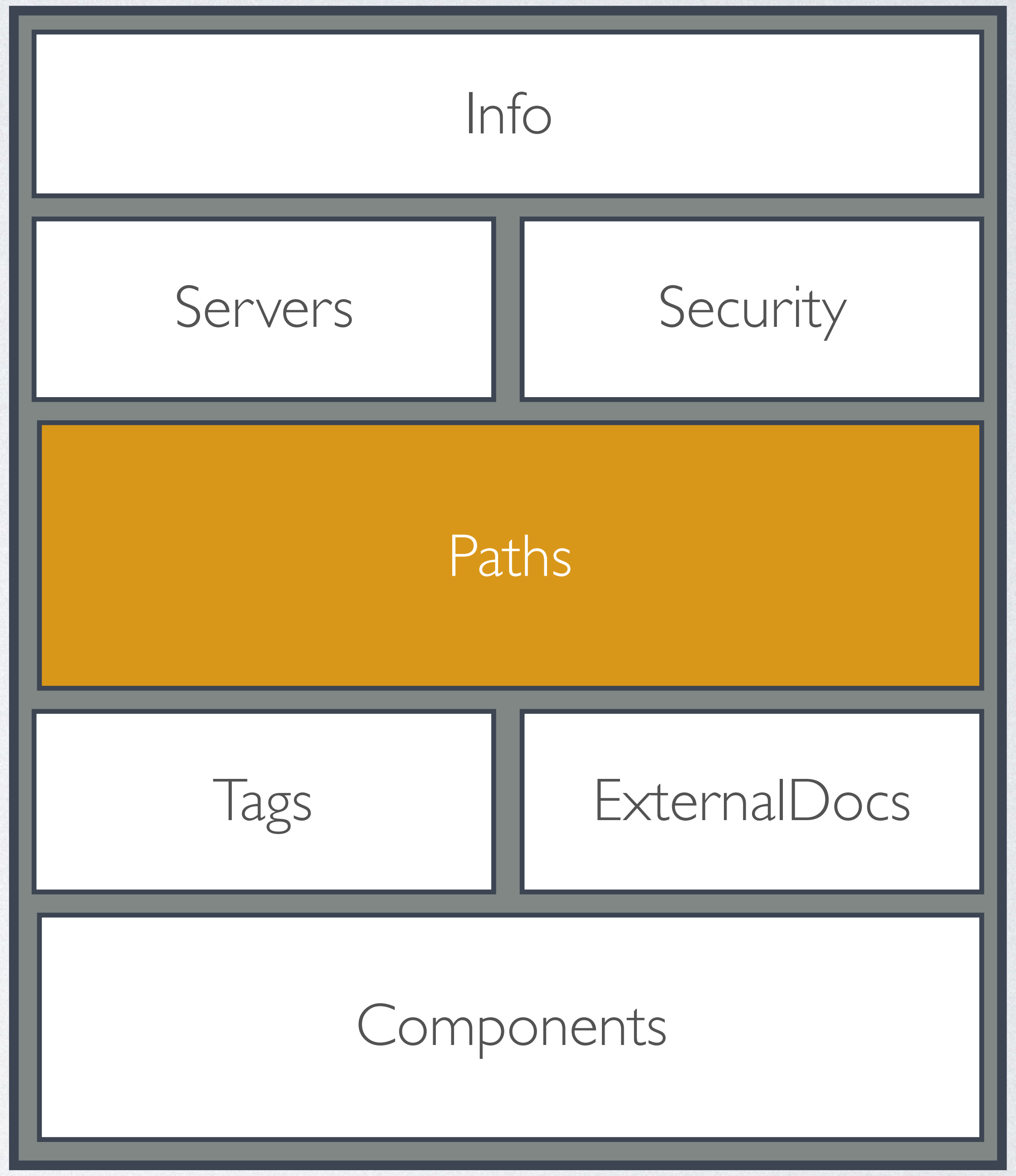
parameters (query, path…)

request body

responses



```yaml
openapi: 3.0.1
info:
  title: Todo Backend
  version: "1.0"
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```



Info

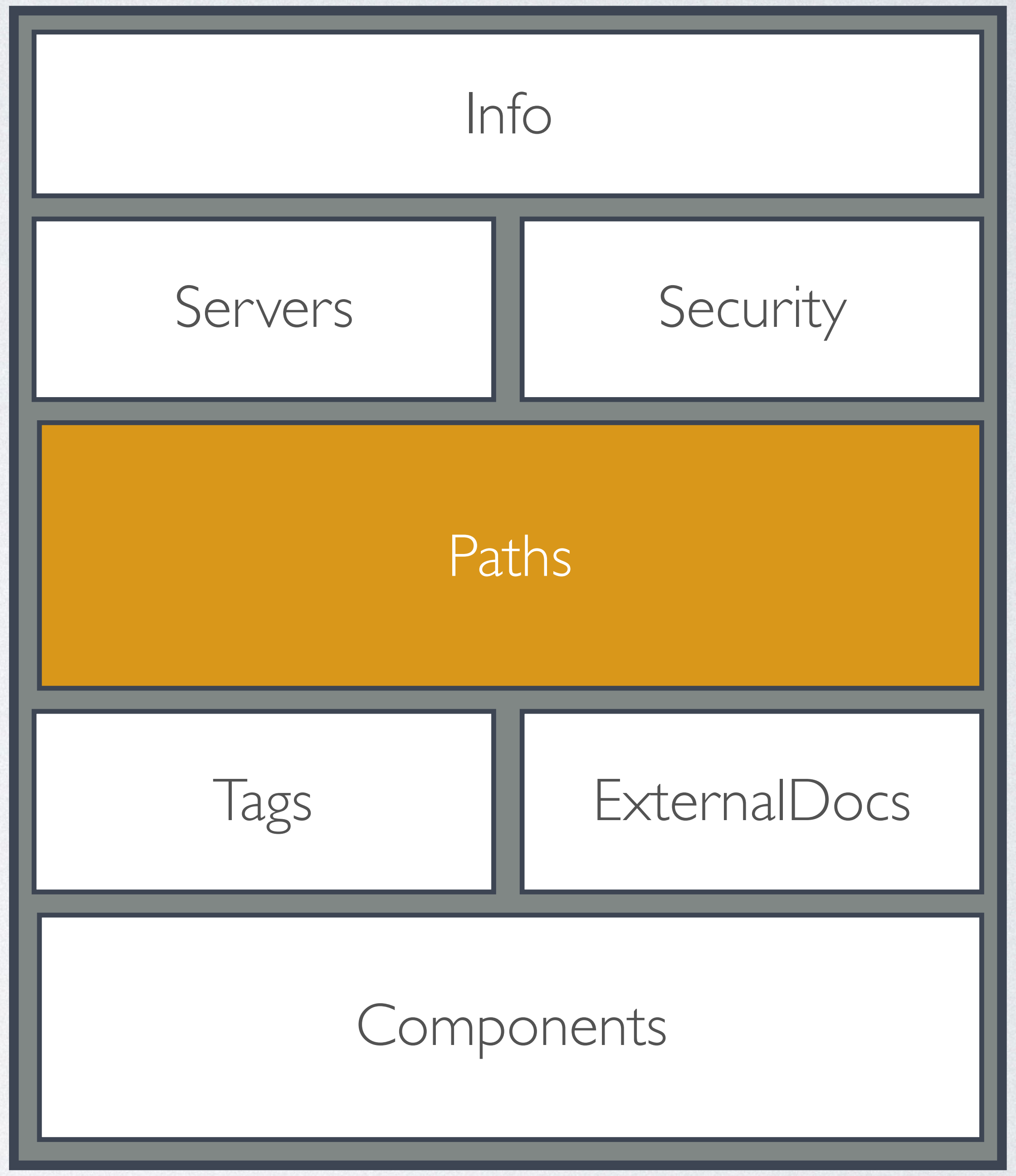
Servers | Security

Paths

Tags | ExternalDocs

Components

```yaml
openapi: 3.0.1
info:
  title: Todo Backend
  version: "1.0"
paths:
  /api/{id}:
    get:
      summary: Get the one todo
      operationId: todoGetOne
      parameters:
      - name: id
        in: path
        description: The id of the todo
        required: true
        schema:
          format: int64
          type: integer
        example: "42"
      responses:
        200:
          description: The requested Todo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/Todo'
```



Info

Servers

Security

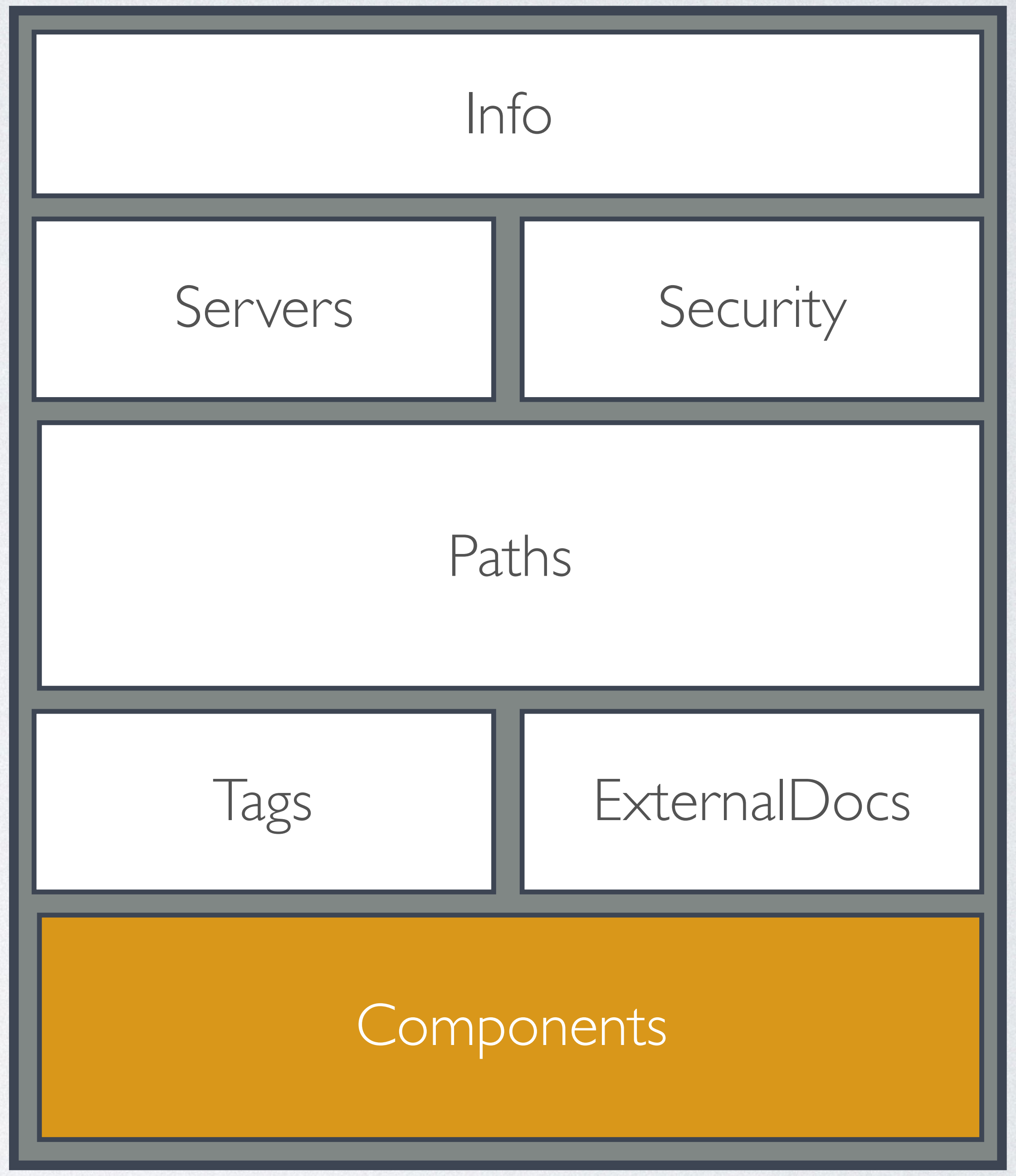
Paths

Tags

ExternalDocs

Components

```yaml
components:
  schemas:
    Todo:
      description: Object representing a Todo
      type: object
      properties:
        id:
          description: id of the entity
          format: int64
          type: integer
          example: "42"
        title:
          description: title of the todo
          type: string
          example: My task
        completed:
          description: whether the todo is completed or not
          type: boolean
          example: "false"
        url:
          description: url associated with the todo
          type: string
        order:
          format: int32
          description: order in the priority list
          type: integer
          example: "10"
```



Info

Servers

Security

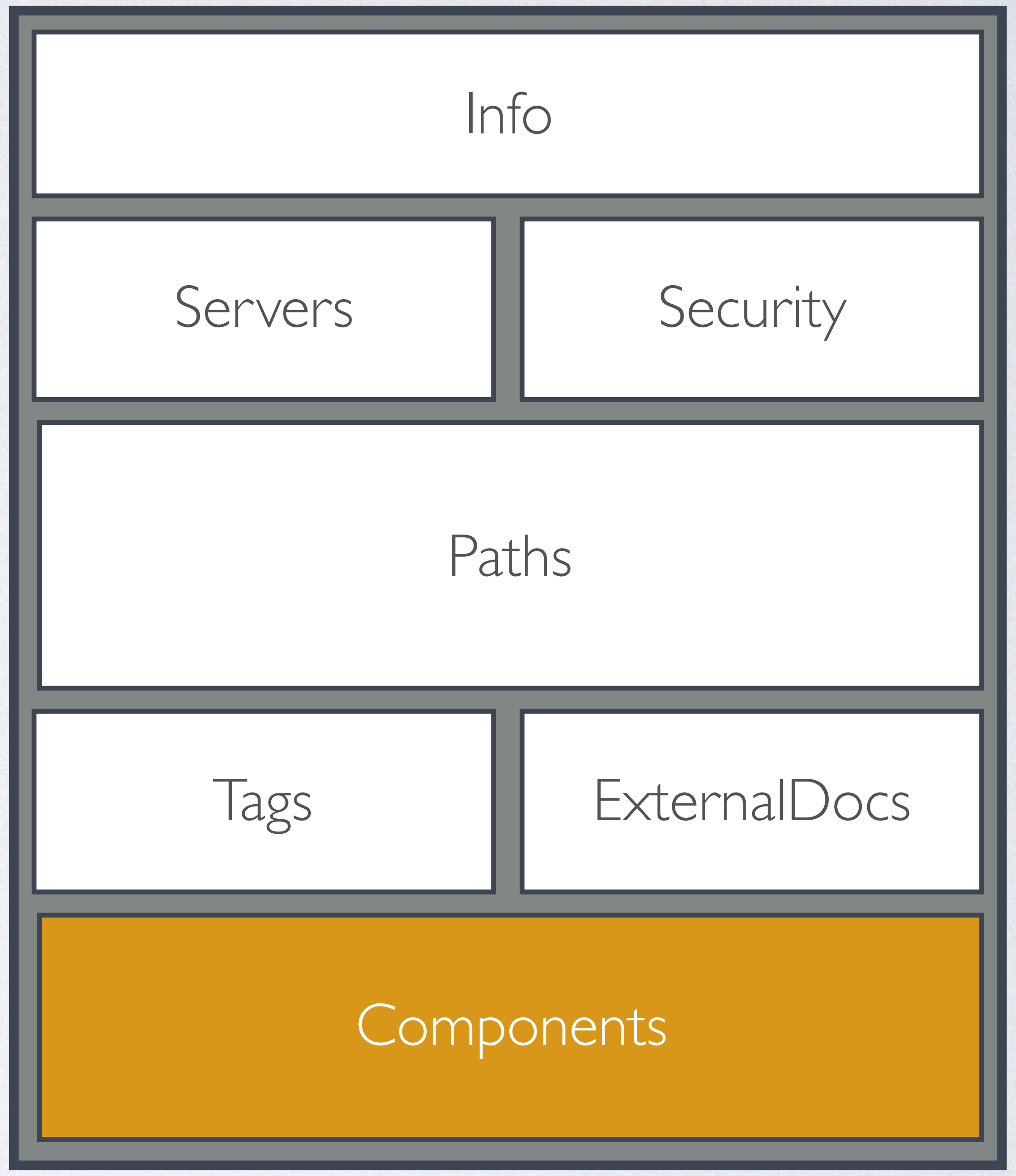
Paths

Tags

ExternalDocs

Components

```yaml
components:
  schemas:
    Todo:
      description: Object representing a Todo
      type: object
      properties:
        id:
          description: id of the entity
          format: int64
          type: integer
          example: "42"
        title:
          description: title of the todo
          type: string
          example: My task
        completed:
          description: whether the todo is completed or not
          type: boolean
          example: "false"
        url:
          description: url associated with the todo
          type: string
        order:
          format: int32
          description: order in the priority list
          type: integer
          example: "10"
```



Info

Servers

Security

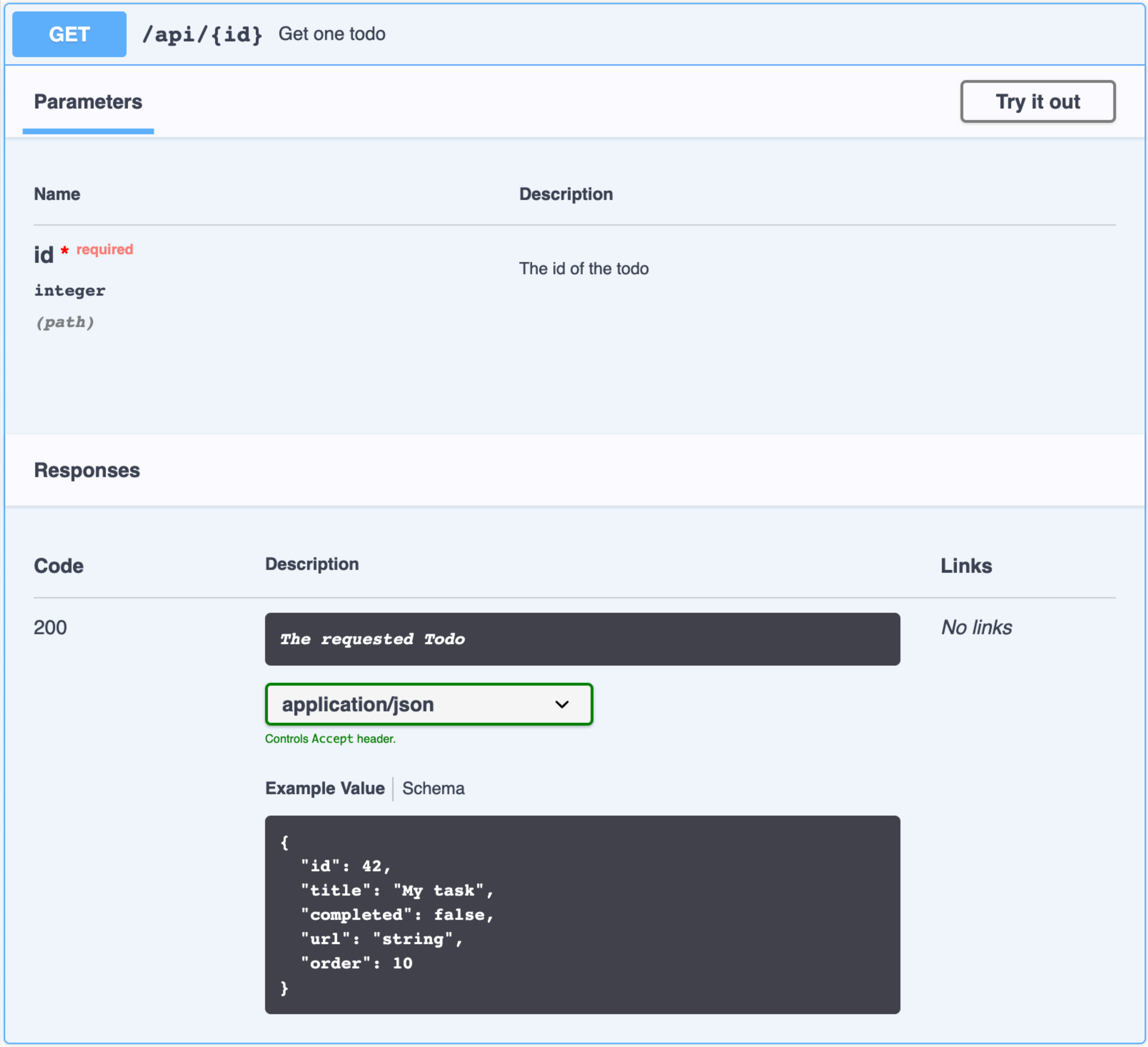
Paths

Tags

ExternalDocs

Components

Swagger UI



`/api/{id}` Get one todo

Parameters

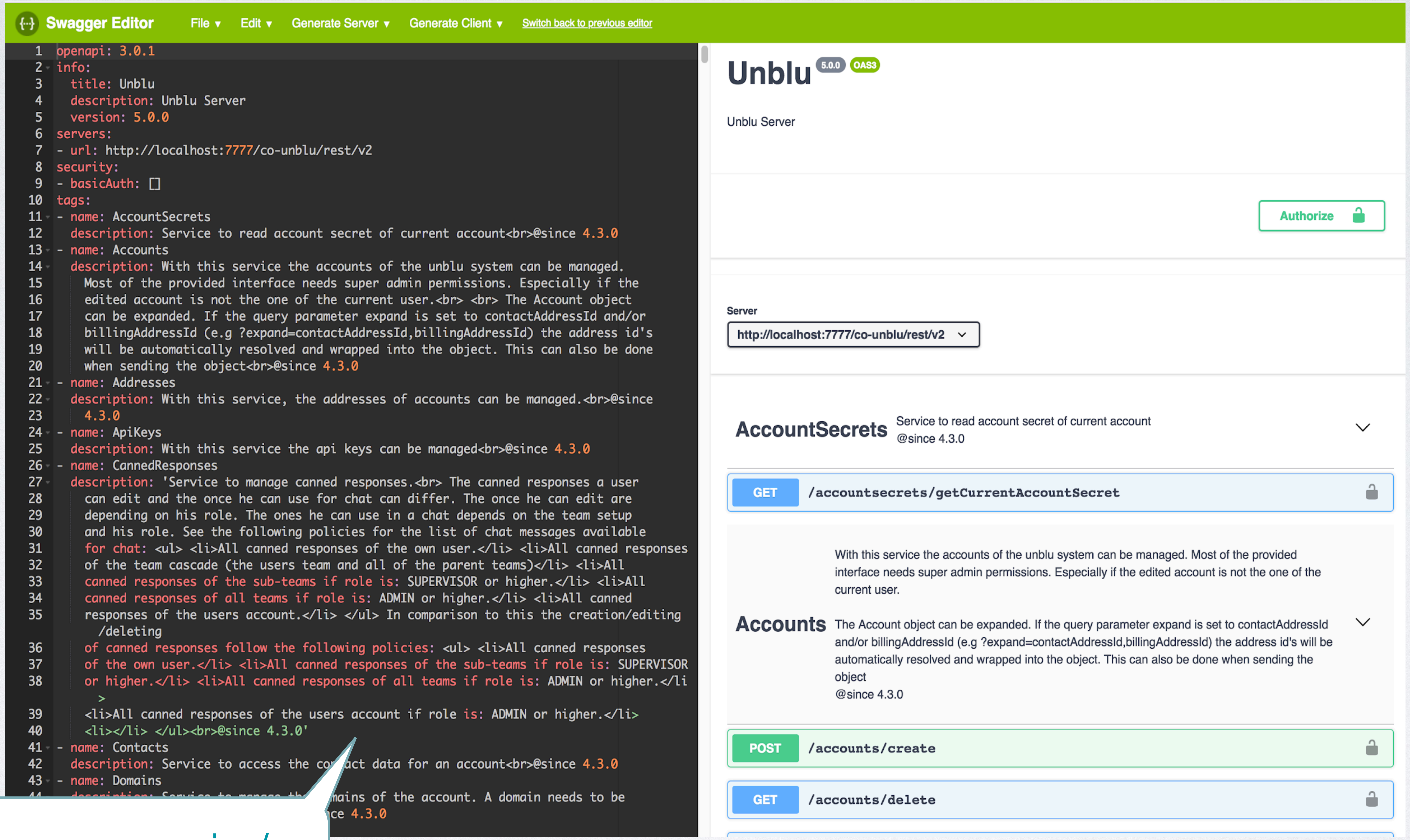
Name	Description
id * required integer *(path)*	The id of the todo

Responses

Code	Description	Links
200	*The requested Todo* application/json Controls Accept header.	*No links*

```
{
  "id": 42,
  "title": "My task",
  "completed": false,
  "url": "string",
  "order": 10
}
```

Swagger Online Editor



```yaml
openapi: 3.0.1
info:
  title: Unblu
  description: Unblu Server
  version: 5.0.0
servers:
- url: http://localhost:7777/co-unblu/rest/v2
security:
- basicAuth: □
tags:
- name: AccountSecrets
  description: Service to read account secret of current account<br>@since 4.3.0
- name: Accounts
  description: With this service the accounts of the unblu system can be managed.
    Most of the provided interface needs super admin permissions. Especially if the
    edited account is not the one of the current user.<br> <br> The Account object
    can be expanded. If the query parameter expand is set to contactAddressId and/or
    billingAddressId (e.g ?expand=contactAddressId,billingAddressId) the address id's
    will be automatically resolved and wrapped into the object. This can also be done
    when sending the object<br>@since 4.3.0
- name: Addresses
  description: With this service, the addresses of accounts can be managed.<br>@since
    4.3.0
- name: ApiKeys
  description: With this service the api keys can be managed<br>@since 4.3.0
- name: CannedResponses
  description: 'Service to manage canned responses.<br> The canned responses a user
    can edit and the once he can use for chat can differ. The once he can edit are
    depending on his role. The ones he can use in a chat depends on the team setup
    and his role. See the following policies for the list of chat messages available
    for chat: <ul> <li>All canned responses of the own user.</li> <li>All canned responses
    of the team cascade (the users team and all of the parent teams)</li> <li>All
    canned responses of the sub-teams if role is: SUPERVISOR or higher.</li> <li>All
    canned responses of all teams if role is: ADMIN or higher.</li> <li>All canned
    responses of the users account.</li> </ul> In comparison to this the creation/editing
      /deleting
    of canned responses follow the following policies: <ul> <li>All canned responses
    of the own user.</li> <li>All canned responses of the sub-teams if role is: SUPERVISOR
    or higher.</li> <li>All canned responses of all teams if role is: ADMIN or higher.</li
      >
    <li>All canned responses of the users account if role is: ADMIN or higher.</li>
    <li></li> </ul><br>@since 4.3.0'
- name: Contacts
  description: Service to access the co  act data for an account<br>@since 4.3.0
- name: Domains
                                       mains of the account. A domain needs to be
                                       ce 4.3.0
```

Unblu

Unblu Server

Server

http://localhost:7777/co-unblu/rest/v2

AccountSecrets Service to read account secret of current account
@since 4.3.0

GET `/accountsecrets/getCurrentAccountSecret`

Accounts With this service the accounts of the unblu system can be managed. Most of the provided interface needs super admin permissions. Especially if the edited account is not the one of the current user.

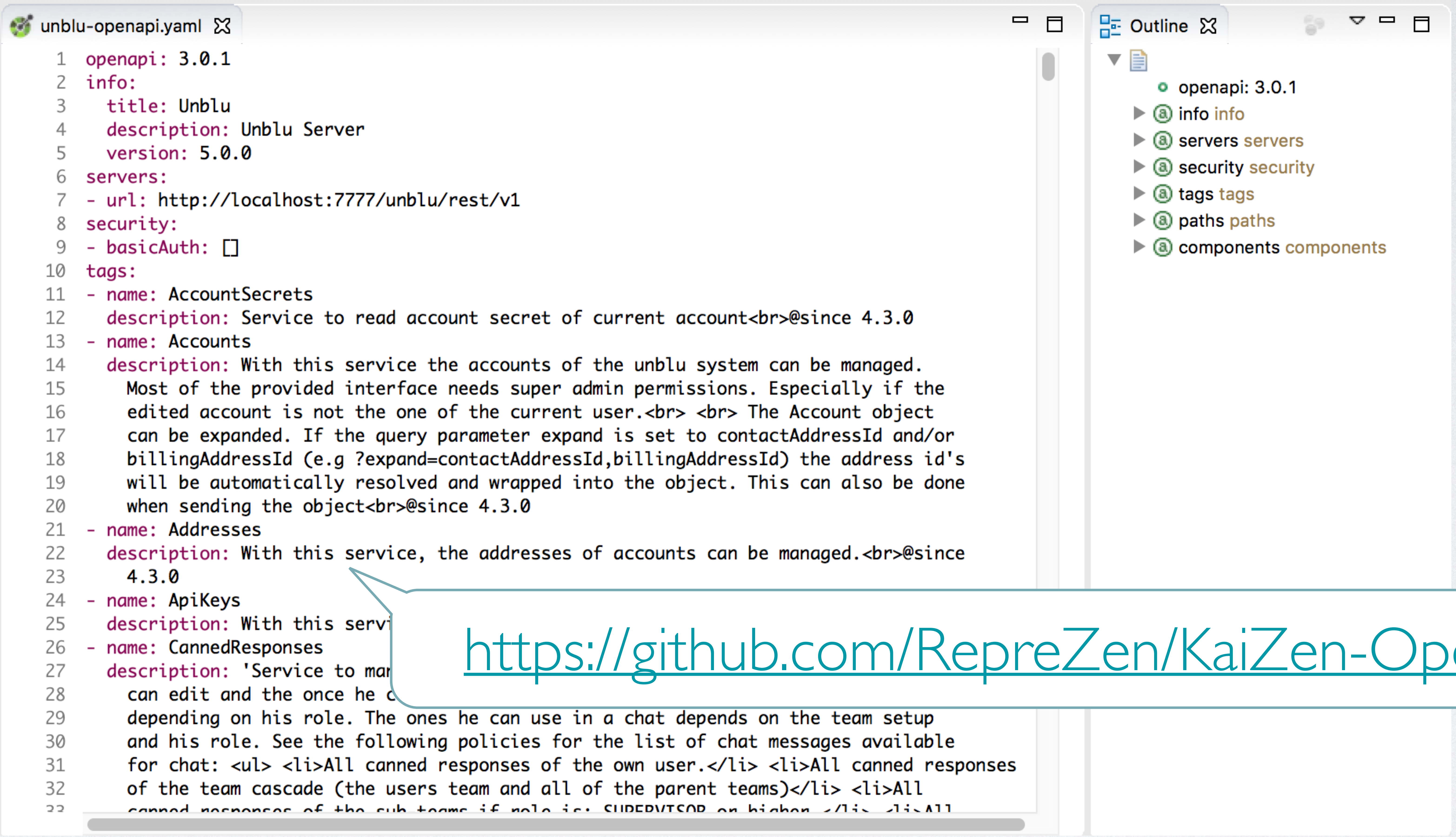
The Account object can be expanded. If the query parameter expand is set to contactAddressId and/or billingAddressId (e.g ?expand=contactAddressId,billingAddressId) the address id's will be automatically resolved and wrapped into the object. This can also be done when sending the object
@since 4.3.0

POST `/accounts/create`

GET `/accounts/delete`

https://editor.swagger.io/

Eclipse IDE plugin: KaiZen-OpenAPI-Editor



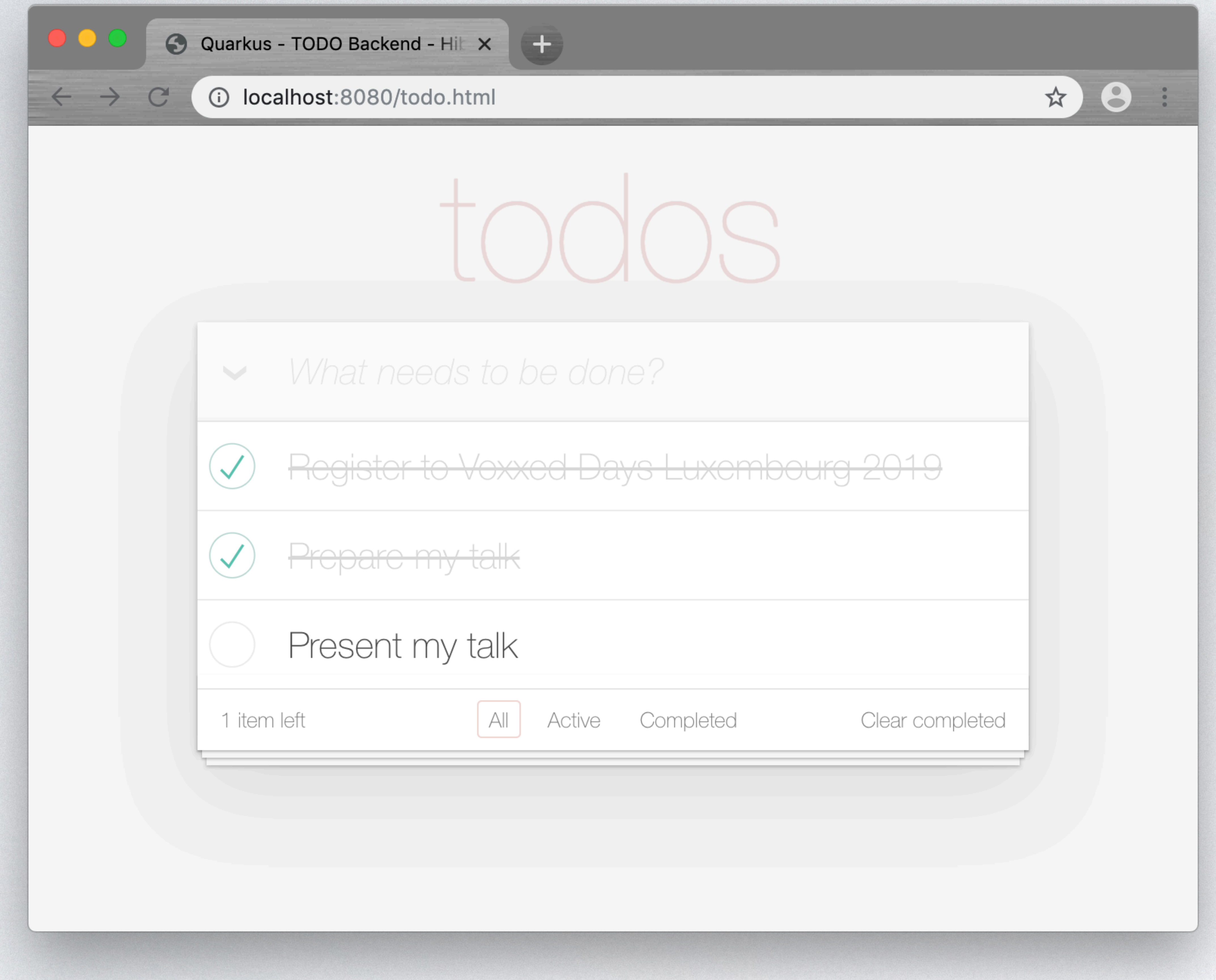
```yaml
openapi: 3.0.1
info:
  title: Unblu
  description: Unblu Server
  version: 5.0.0
servers:
- url: http://localhost:7777/unblu/rest/v1
security:
- basicAuth: []
tags:
- name: AccountSecrets
  description: Service to read account secret of current account<br>@since 4.3.0
- name: Accounts
  description: With this service the accounts of the unblu system can be managed.
    Most of the provided interface needs super admin permissions. Especially if the
    edited account is not the one of the current user.<br> <br> The Account object
    can be expanded. If the query parameter expand is set to contactAddressId and/or
    billingAddressId (e.g ?expand=contactAddressId,billingAddressId) the address id's
    will be automatically resolved and wrapped into the object. This can also be done
    when sending the object<br>@since 4.3.0
- name: Addresses
  description: With this service, the addresses of accounts can be managed.<br>@since
    4.3.0
- name: ApiKeys
  description: With this servi
- name: CannedResponses
  description: 'Service to mar
    can edit and the once he c
    depending on his role. The ones he can use in a chat depends on the team setup
    and his role. See the following policies for the list of chat messages available
    for chat: <ul> <li>All canned responses of the own user.</li> <li>All canned responses
    of the team cascade (the users team and all of the parent teams)</li> <li>All
    canned responses of the sub teams if role is: SUPERVISOR or higher.</li> <li>All
```

- openapi: 3.0.1
- info info
- servers servers
- security security
- tags tags
- paths paths
- components components

https://github.com/RepreZen/KaiZen-OpenAPI-Editor

DEMO

Demo: todo-backend



Quarkus - TODO Backend - Hi

localhost:8080/todo.html

todos

What needs to be done?

✓ ~~Register to Voxxed Days Luxembourg 2019~~

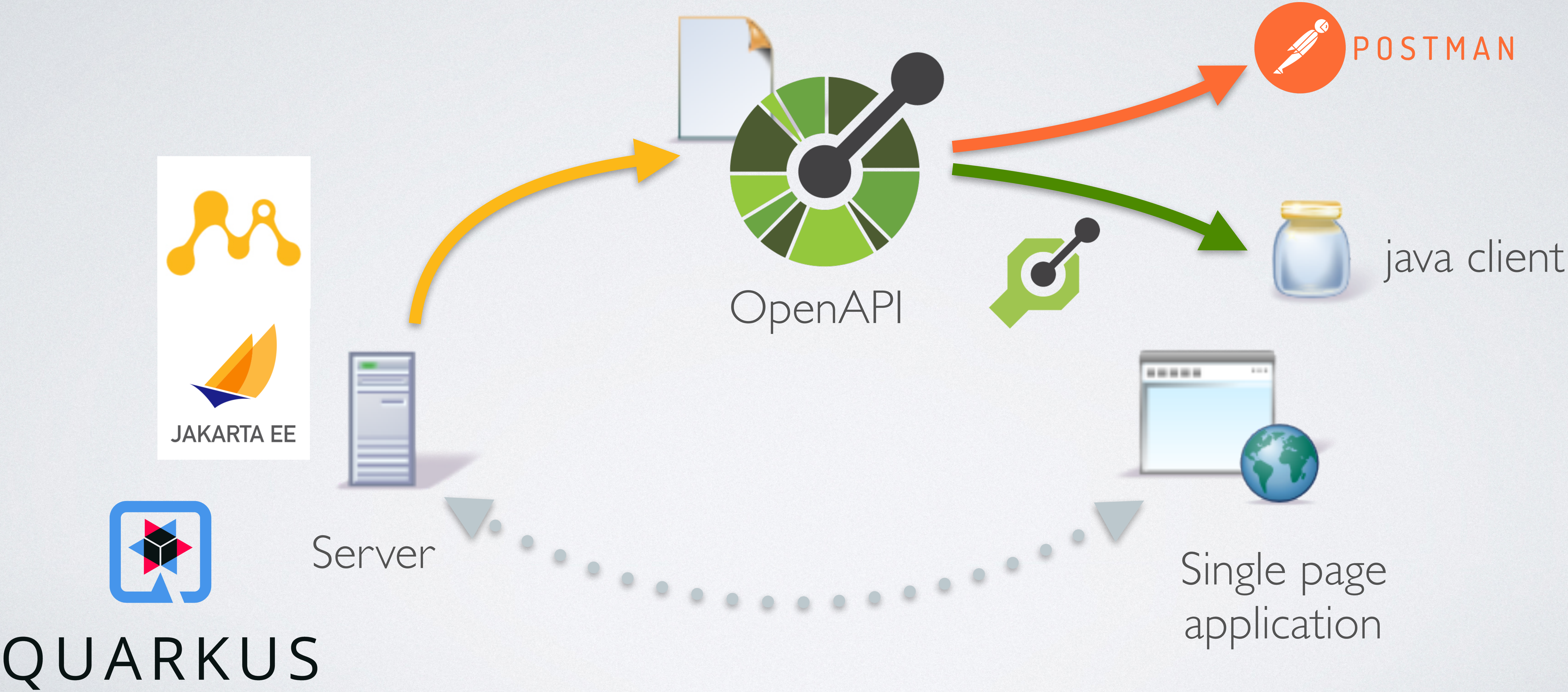
✓ ~~Prepare my talk~~

○ Present my talk

1 item left All Active Completed Clear completed

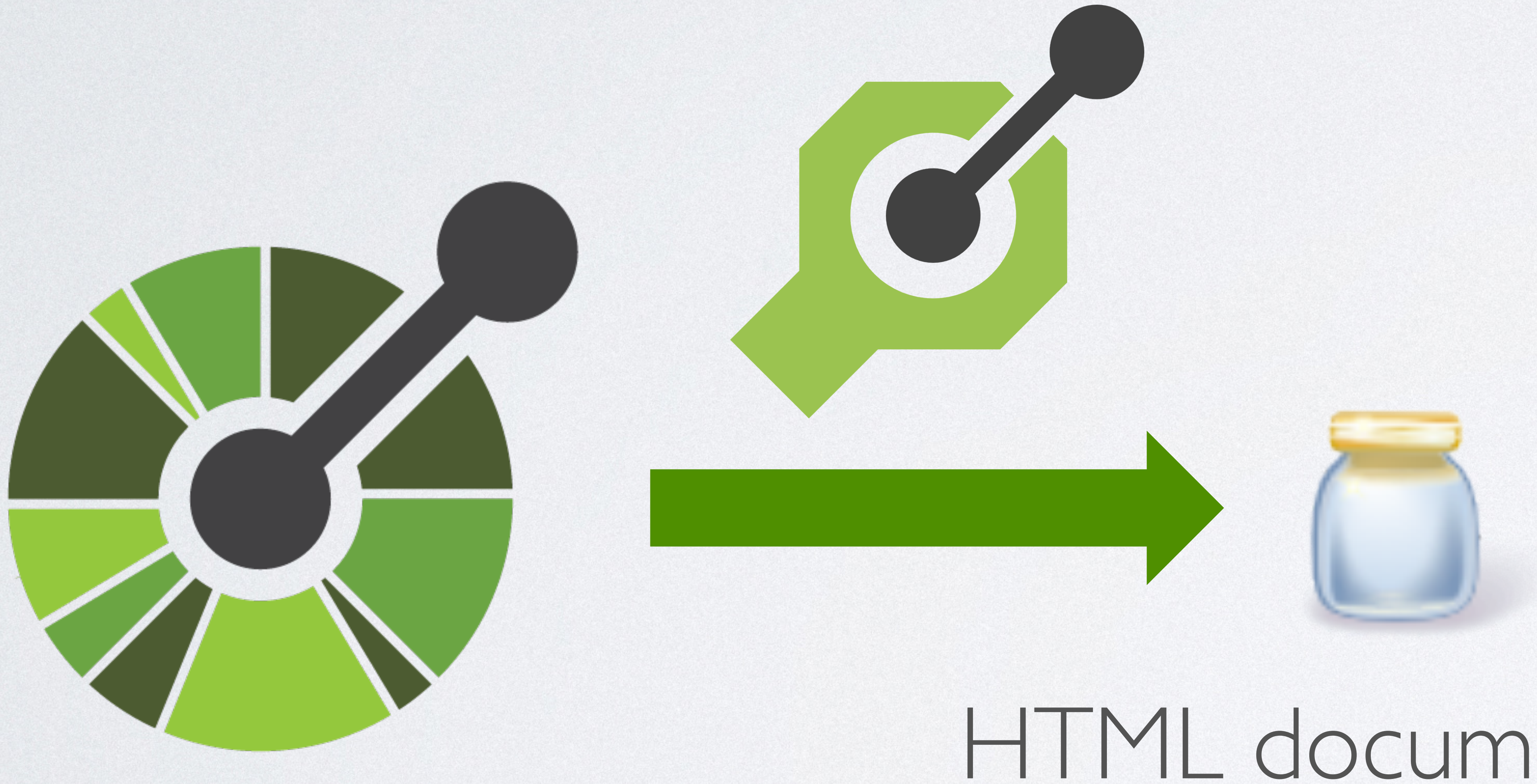
QUARKUS

Code first approach



DEMO

Code generator: OpenAPI-Generator

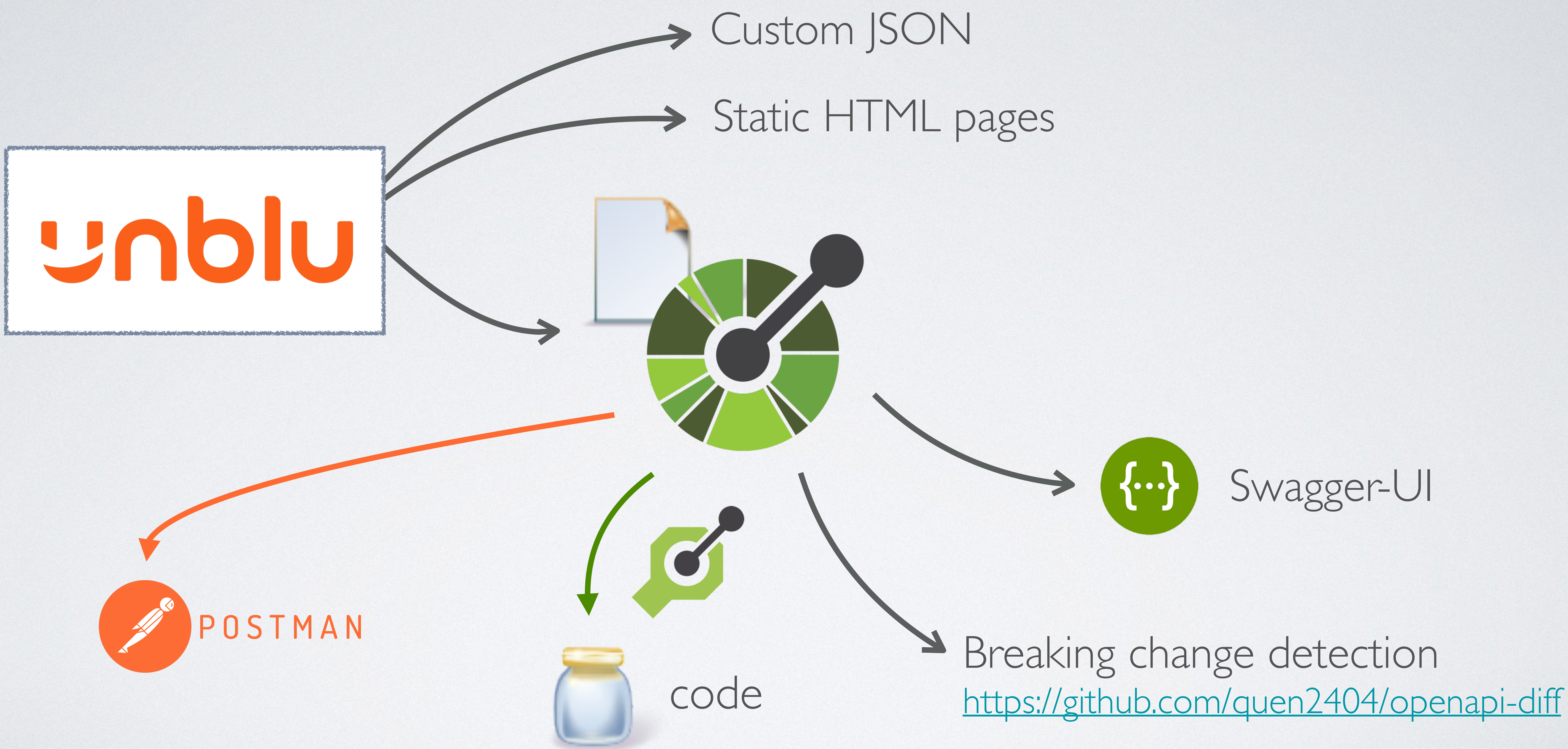


OpenAPI Generator

OpenAPI
Specification

HTML documentation
Client Code
Server Stub
...

- **Open Source** (Apache 2.0 License)
- Hosted on **GitHub**: https://github.com/OpenAPITools/openapi-generator
- Java code & mustache templates
- Fork of Swagger-Codegen



Custom JSON

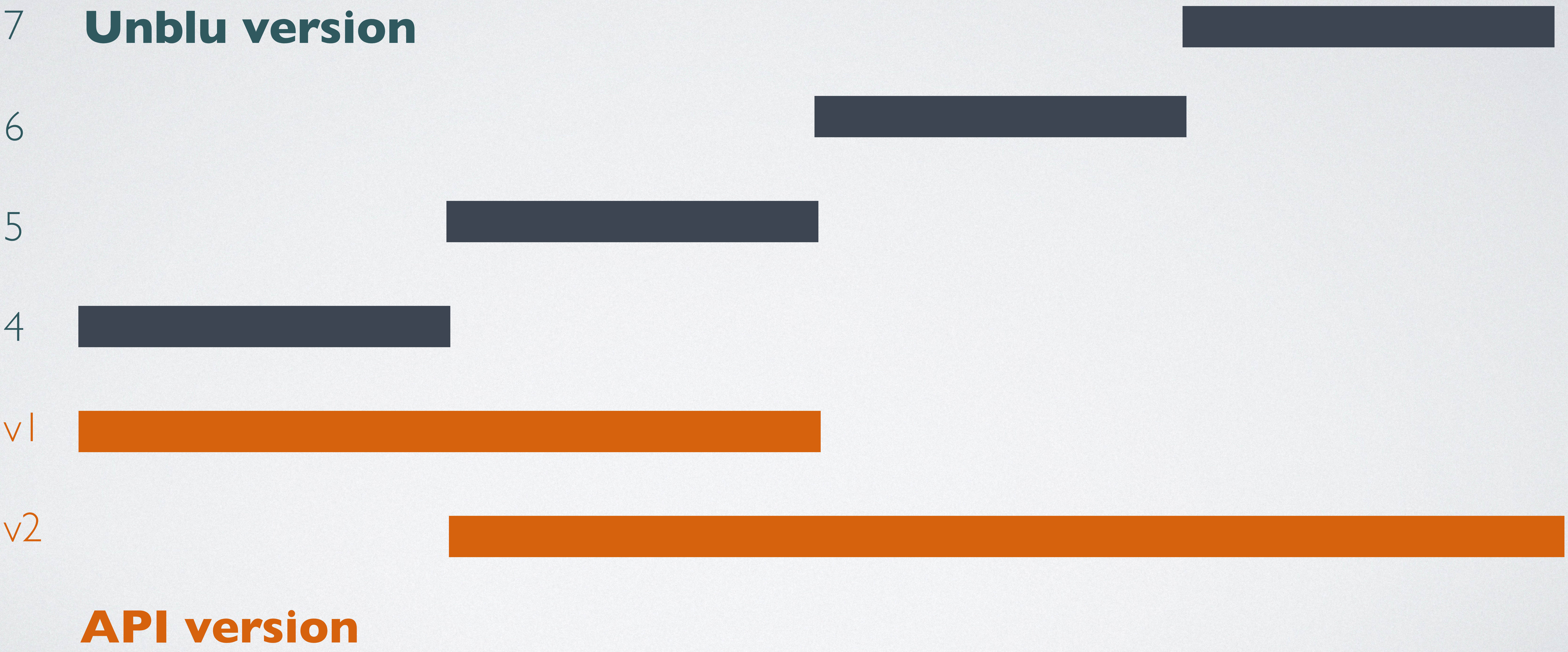
Static HTML pages

Swagger-UI

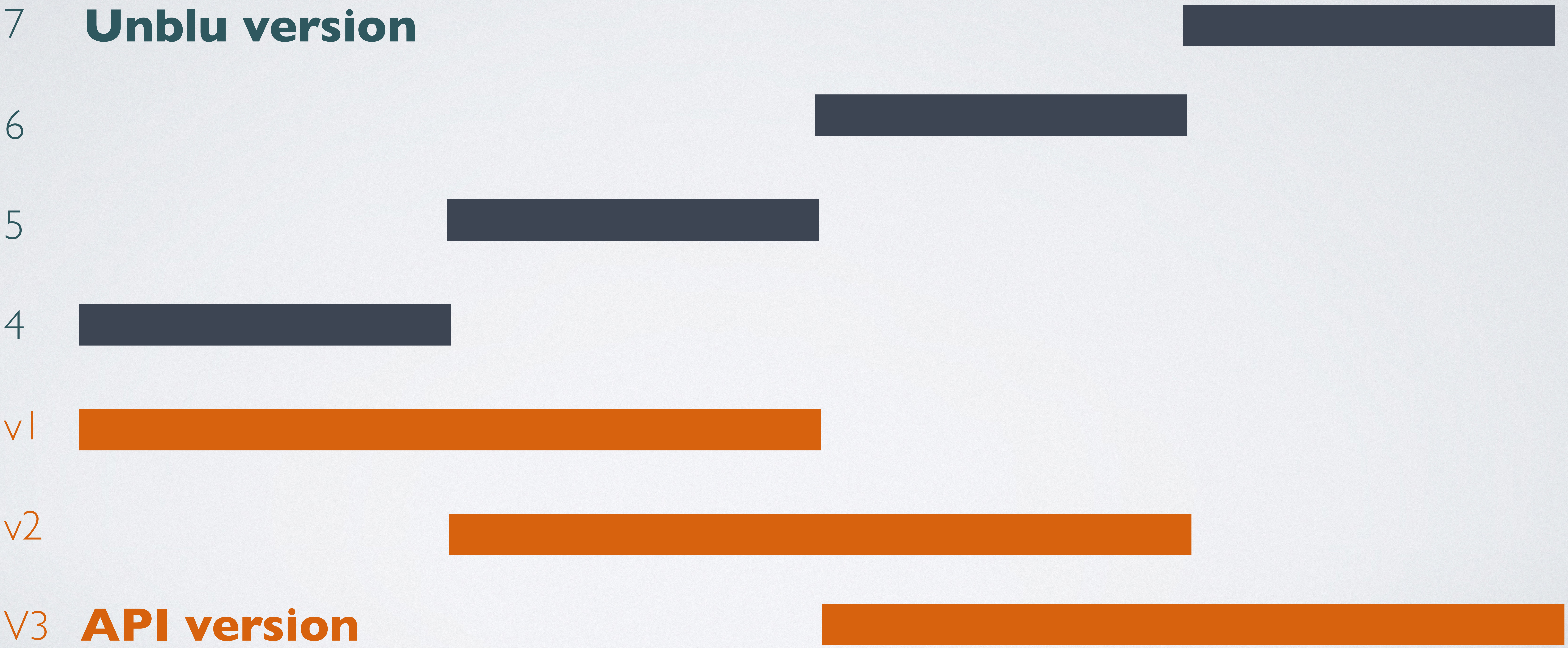
code

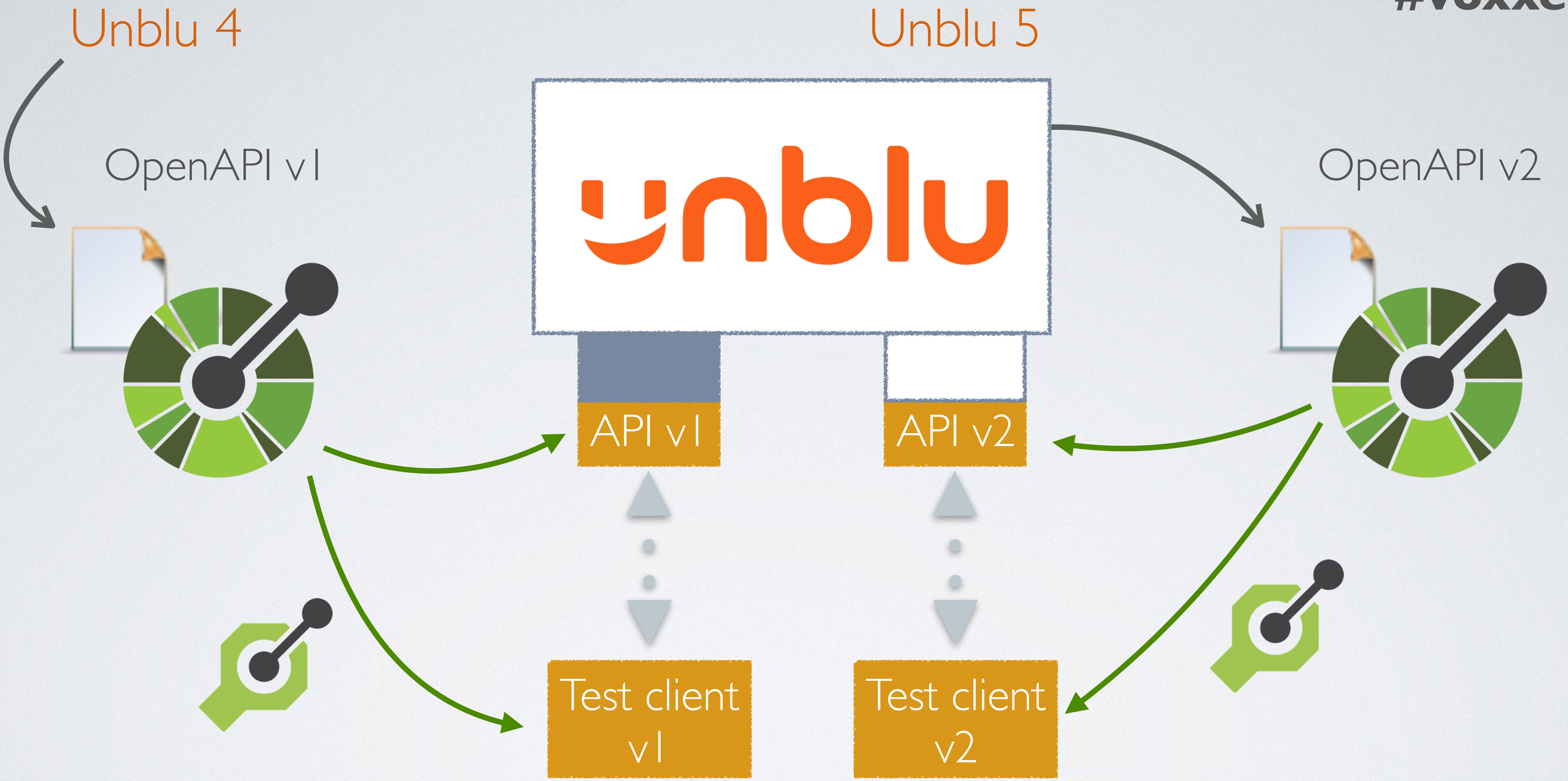
Breaking change detection
https://github.com/quen2404/openapi-diff

API versioning



API versioning





Unblu 4

OpenAPI v1

Unblu 5

OpenAPI v2

API v1

API v2

Test client v1

Test client v2

Thank you!

@j2r2b

jmini

Code Examples: https://github.com/jmini/openapi-talk