



API-first!
Contract-first!
Make API development
great again! v2.0



Atanas L. Kuzmanov
Senior Software Engineer

Hello, my name is

Atanas L. Kuzmanov
Senior Software Engineer
IT Solution Development
SchwarzIT Bulgaria

Background:

- Where do you work?
- What do you do there?
- Why are you talking about this topic?
- Why are you giving this talk today?

About me:

Previously:



accenture



Currently:



{ <https://github.com/atkuzmanov>
<https://www.linkedin.com/in/atkuzmanov/>
<https://twitter.com/atkuzmanov>
<https://linktr.ee/atkuzmanov>
<https://atkuzmanov.wordpress.com/> }



Agenda

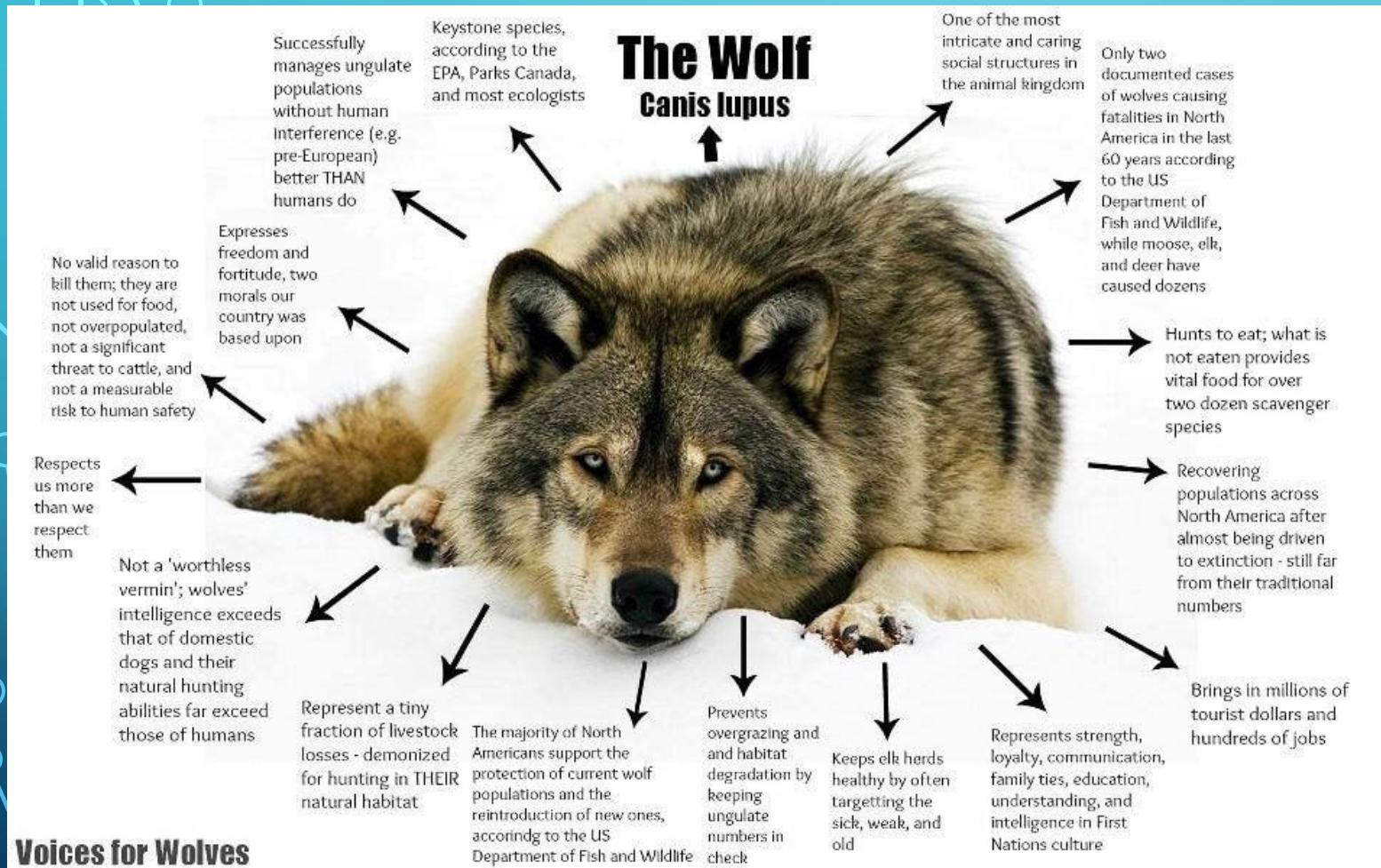
- 1 INTRODUCTION
- 2 DESIGN FIRST VS. CODE FIRST
- 3 CODE FIRST
- 4 DESIGN FIRST APPROACH OR API FIRST APPROACH
- 5 YOUR API IS YOUR CONTRACT
- 6 CONTRACT TESTING AND VALIDATION
- 7 TOOLING
- 8 DEMO Part 1
- 9 CLASH OF THE TITANS - Spring REST and JAX-RS
- 10 DEMO Part 2
- 11 CONCLUSIONS
- 12 CLOSING THOUGHTS
- 13 REFERENCES
- 14 QUESTIONS



INTRODUCTION

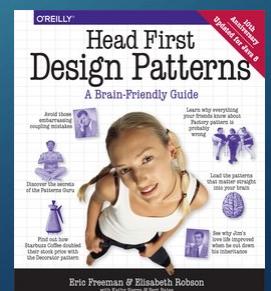
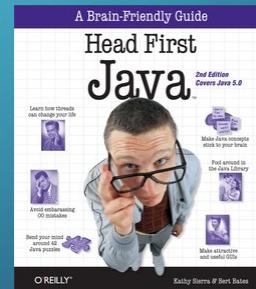


SIDE NOTE



eurONATUR STIFTUNG

<https://www.euronatur.org/en/what-we-do/endangered-species/wolf/wolves-in-europe/>





API



<https://youtu.be/yqrZMEraAJ8>



https://youtu.be/FW-m_j4yINU



https://youtu.be/t_mjnH-vOaQ



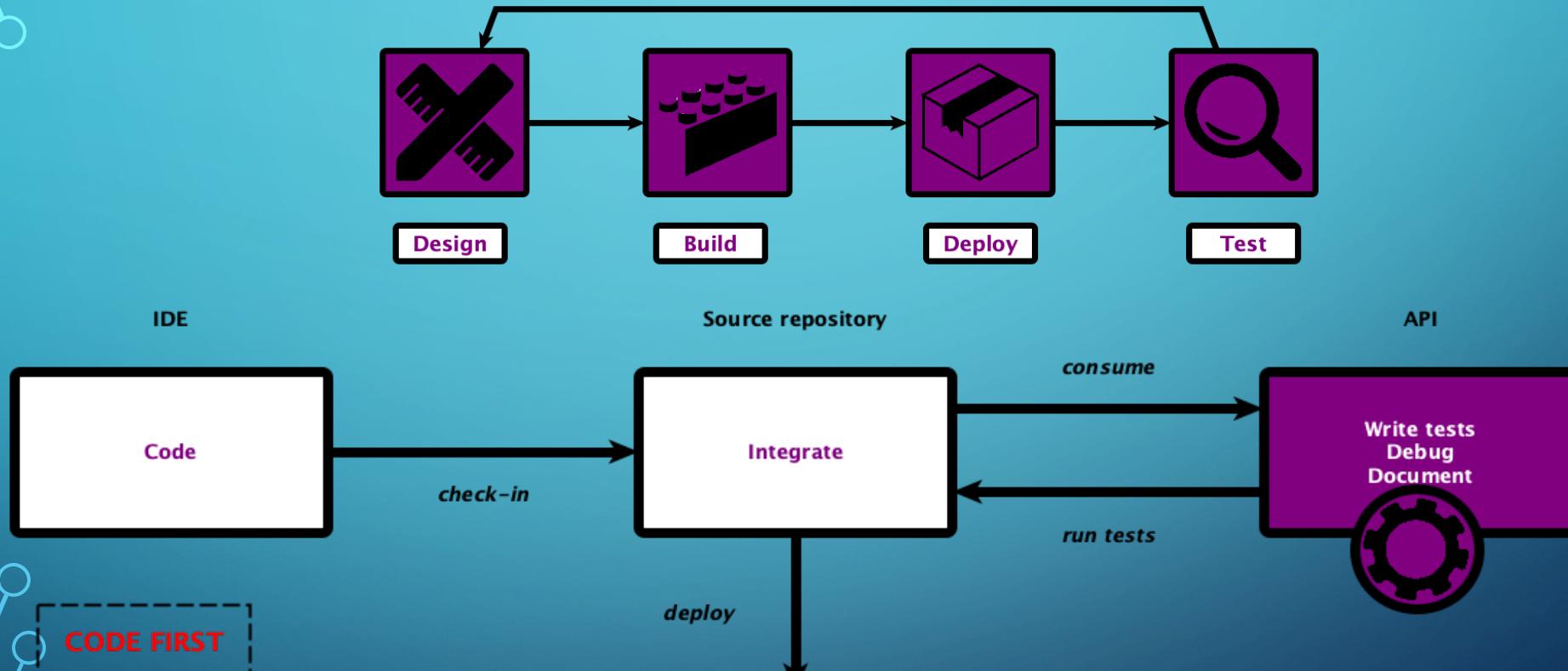


DESIGN FIRST
VS.
CODE FIRST



CODE FIRST

WHEN THE CODE COMES FIRST





CHALLENGES OF THE CODE FIRST APPROACH

- Code First doesn't mean leaving out the design.
- Rather it means the design is hidden away in Jira tickets, Confluence pages, text documents... napkins... and potentially lost to history...
- This can lead to several problems.
- Frustrated Users
- Building the Wrong API
- Missed Opportunities



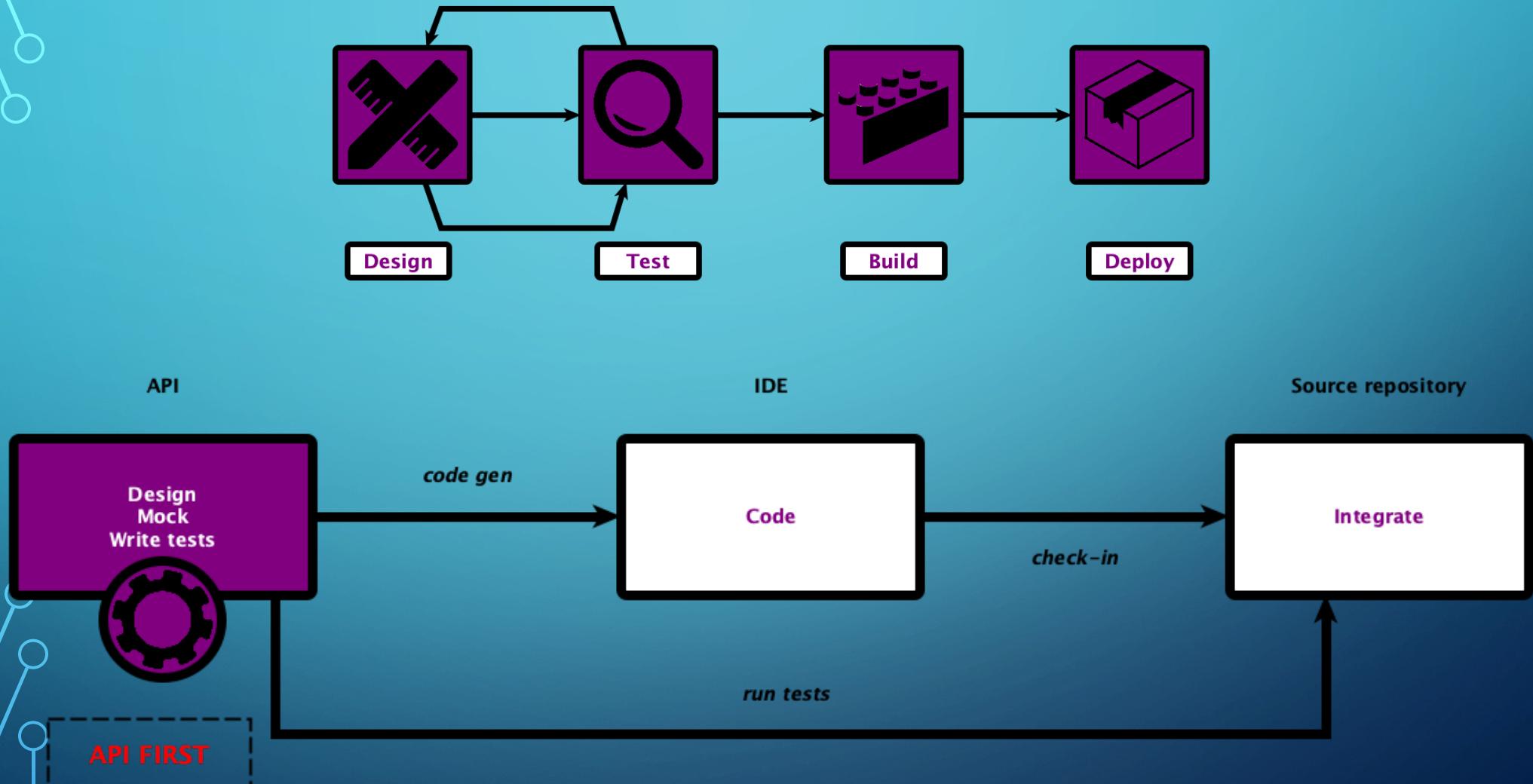
BENEFITS OF THE CODE FIRST APPROACH

- ***When Delivery Speed Matters*** - Developers can start coding directly from the requirements.
- ***When Developing Internal APIs*** - If the API will only be consumed by the team or company that's building it, and is small, then the Code First approach is an ideal solution.



DESIGN FIRST APPROACH
OR
API FIRST APPROACH

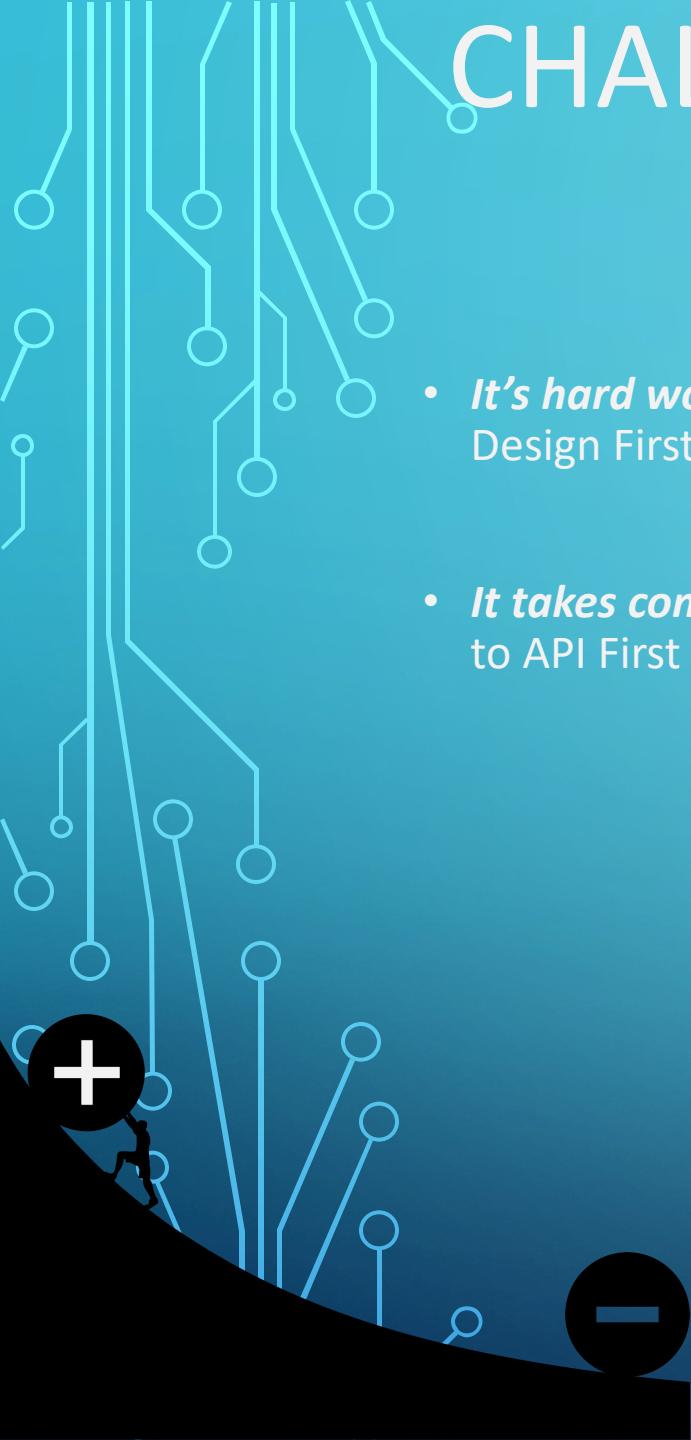
WHEN THE API COMES FIRST



-

CHALLENGES OF THE DESIGN FIRST APPROACH

- ***It's hard work*** - To be honest, it takes a lot of hard work for any team to move to a Design First approach.
- ***It takes commitment*** - Companies must be intentional no matter their size - this shift to API First doesn't happen on its own and it doesn't happen overnight.





WHY API FIRST? BENEFITS OF DESIGN FIRST

The API First approach gives us the following main advantages:

- *It allows early feedback from stakeholders with fast turnaround cycles.*
- *It happens at a stage when changes are easily doable, before a lot of time and effort are sunk into the project.*
- *It leads to Product-Driven APIs with better user experience and higher customer satisfaction.*
- *It reduces coupling and enables provider and consumer teams to work independently and in parallel.*
- *The API definition can be the Single Source of Truth for Developers, DevOps and Architects.*

{API}

REAL WORLD EXAMPLES



SCHWARZ



WOLF FACT



True love

- Once a wolf has found a mate, they tend to stay together for better or worse, through sickness and health, often until death due them part.
- They have been known to sacrifice themselves for the survival of other wolves.
- So, move over Romeo and Juliet!



YOUR API IS YOUR CONTRACT



WHAT IS AN API CONTRACT?

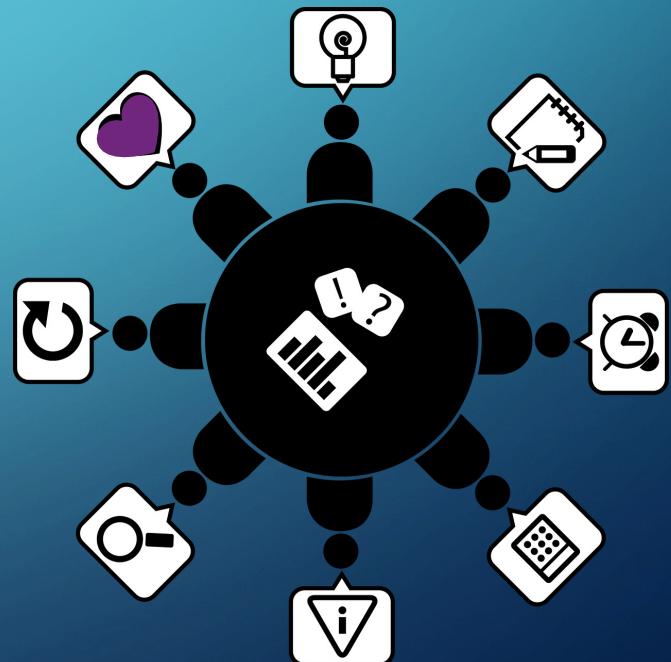


- API First means that implementation starts with the definition of an API.
- The API defines the contract for providers and consumers to rely on, i.e. the interface is clearly defined.
- The contract uses tools such as OpenAPI to describe the requests and responses of each API.
- It is something that both API provider and consumer agree on, and get to work on delivering and consuming - for an agreed upon price.
- An API contract helps communicates change.
- Providing an OpenAPI spec before each new version allows consumers to review the contract before they commit to it.



WHAT IS AN API CONTRACT?

- This sounds great and all, but if there is no way to enforce a contract, people might not keep it?
- This can have disastrous consequences.
- If only there was a way to change this?!?



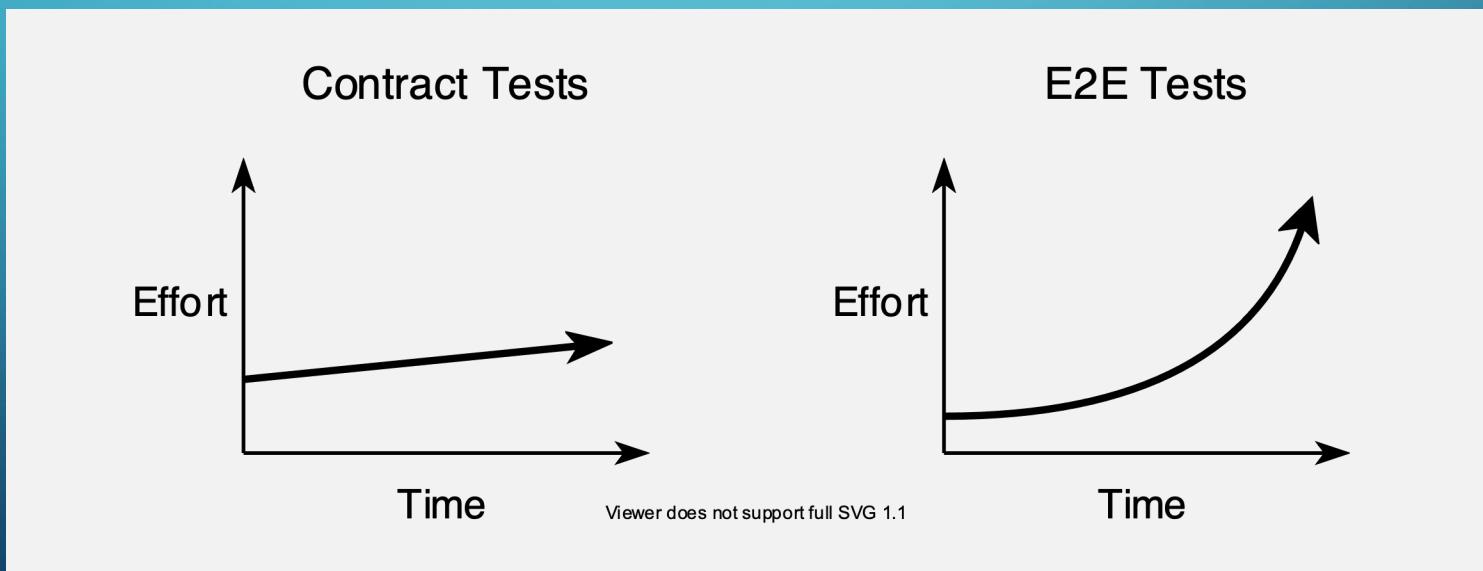


CONTRACT TESTING AND VALIDATION



WHAT IS THIS CONTRACT TESTING THING?

- This is the way we enforce and validate our contracts.
- When compared to E2E testing, contract testing enables you to create consistent curves in your pipeline speed and complexity that look more like the chart below.
- And still maintain many of the safety guarantees of E2E tests.





CONVINCE ME

- The main thing that any organisation cares about when it comes to software development is cost.
- When building a product, your time is probably the biggest cost to an organisation.
- There will be initial overhead in learning how to use and integrate contract tests in your pipelines.
- Using contract tests in conjunction with, or as a replacement for traditional end-to-end tests will be cheaper in the long run.



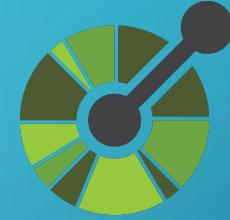
TOOLING

OPENAPI SPEC ANATOMY

- *Main body YAML*
- *Schemas YAML*

```
components:  
  schemas:  
    Pet:  
      type: object  
      required:  
        - id  
        - name  
      properties:  
        id:  
          type: integer  
          format: int64  
        name:  
          type: string  
        tag:  
          type: string
```

```
openapi: "3.0.0"  
info:  
  version: 1.0.0  
  title: Swagger Petstore  
  license:  
    name: MIT  
servers:  
  - url: http://petstore.swagger.io/v1  
paths:  
  /pets:  
    get:  
      summary: List all pets  
      operationId: listPets  
      tags:  
        - pets  
      parameters:  
        - name: limit  
          in: query  
          description: How many items to return at one time (max  
100)  
          required: false  
          schema:  
            type: integer  
            format: int32  
      responses:  
        '200':  
          description: A paged array of pets  
          headers:  
            x-next:  
              description: A link to the next page of responses  
              schema:  
                type: string  
          content:  
            application/json:  
              schema:  
                $ref: "#/components/schemas/Pets"  
        default:  
          description: unexpected error  
          content:  
            application/json:  
              schema:  
                $ref: "#/components/schemas/Error"
```





SWAGGER

The image shows a screenshot of the Swagger Editor interface. On the left, there is a code editor window displaying a Swagger 2.0 JSON schema for a Petstore server. The schema includes definitions for 'pet' and 'store' resources, with various methods like POST /pet, PUT /pet, GET /pet/{petId}, etc. On the right, the main browser window displays the 'Swagger Petstore' homepage at version 1.0.0. It features a navigation bar with links for Terms of service, Contact the developer, Apache 2.0, and Find out more about Swagger. Below this is a 'Schemes' dropdown set to 'HTTPS'. The main content area shows the 'pet' resource with several operations listed: POST /pet, PUT /pet, GET /pet/findByStatus, GET /pet/findByTags, GET /pet/{petId}, POST /pet/{petId}, DELETE /pet/{petId}, and POST /pet/{petId}/uploadImage. The 'store' resource is also partially visible. A green 'Authorize' button is located in the top right corner of the main content area.

“Code-first
and meh
docs later...”

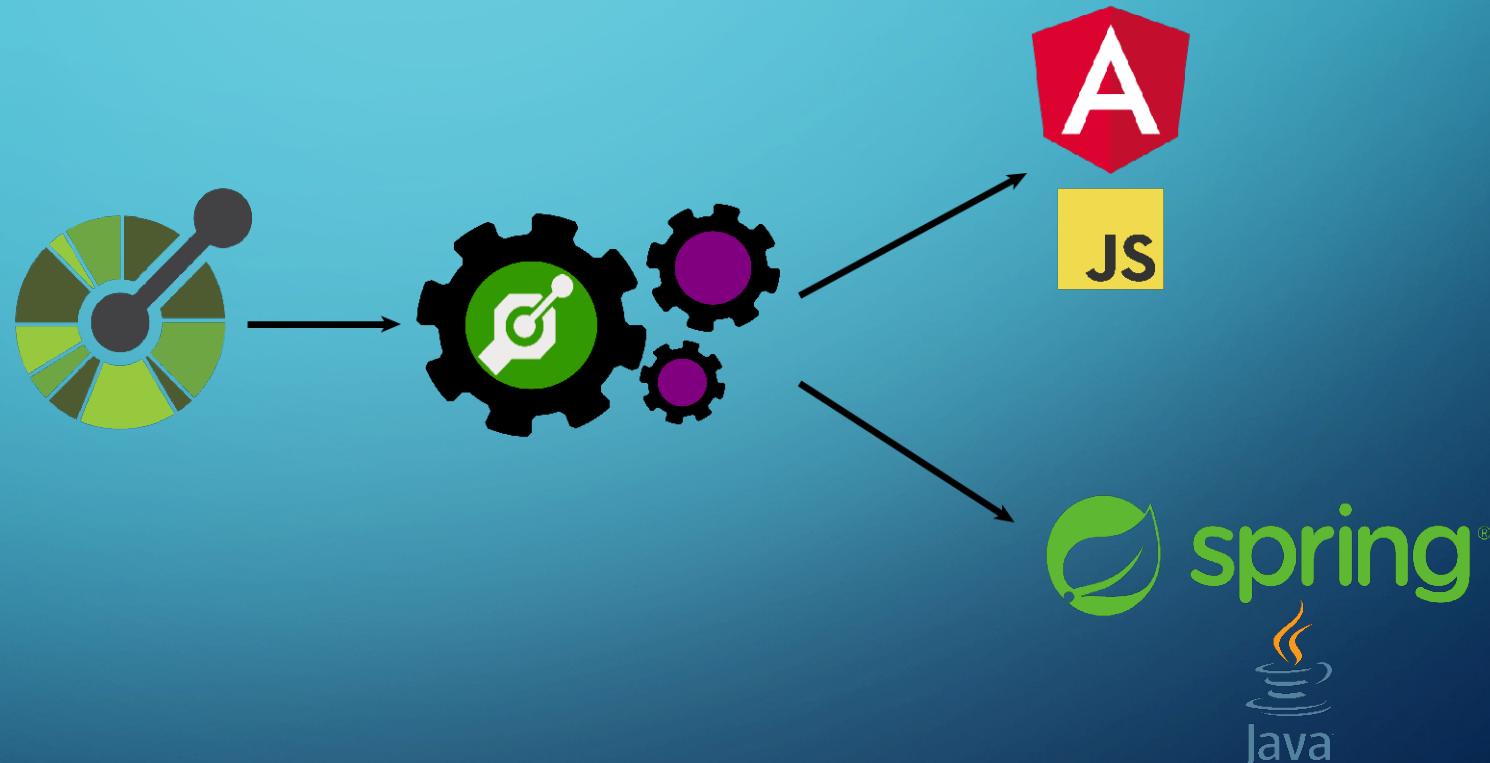
We will write docs
“when we have
time”... almost
definitely for sure...
maybe...”



OPENAPI GENERATOR: SPEC-TO-CODE GENERATION

Given an OpenAPI Spec, and using an *OpenAPI Generator* you can generate:

- API clients.
- Server stubs.
- Documentation.
- Etc.





OPENAPI GENERATOR: CODE-TO-SPEC GENERATION

Given an existing code base, and using an *OpenAPI Generator*, generate an OpenAPI Spec.





ATLASSIAN OPENAPI VALIDATOR

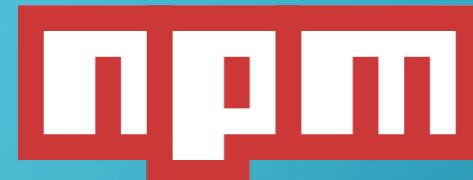
After spending a bit of time researching how can I do request and response validation for our existing APIs, without having to write a library to do the work, I found the answer!

Atlassian – Request Validator

- A Java library for validating HTTP request/responses against an OpenAPI specification.
- It has several modules for different frameworks including integrations with the REST Assured testing library.



FRONTEND OPENAPI VALIDATOR



And here are a couple of OpenApi validators for the frontend, using ExpressJS, which automatically validate API requests and responses based on an OpenAPI spec.

- *express-openapi-validate* (<https://www.npmjs.com/package/express-openapi-validate>)
- *express-openapi-validator* (<https://www.npmjs.com/package/express-openapi-validator>)

OTHER TOOLS

- *Java*
- *SpringBoot*
- *JavaScript*
- *TypeScript*
- *Angular*
- *MongoDB*
- *Docker*





WOLF FACT



The Wolf /Canis Lupus/

- Represents strength, loyalty, family ties, caring and expresses freedom and fortitude.
- The Japanese word for wolf, "Ookami", means "Great God".



DEMO PART 1



DEMO PART 1 TODO

- Look at an OpenAPI Spec.
- Use it and an `openapi-generator` to generate a back-end server in Java and SpringBoot.
- Use it and an `openapi-generator` to generate a front-end client in JavaScript and Angular.
- Use Atlassian's validator to add tests to the back-end server and validate our API.
- Use `express-openapi-validate` to add tests to the front-end server and validate our API.
- Demonstrate how the tests fail if we breach our API contract.



{DEMO}

{DEMO RECAP}

Contract Validation Demo links:

- **Demo Video Part 1:**

<https://www.youtube.com/watch?v=KGgqdpDWh0I>

<https://youtu.be/KGgqdpDWh0I>

- **Demo Video Part 2:**

<https://www.youtube.com/watch?v=0AvPftcZvWY>

<https://youtu.be/0AvPftcZvWY>

- **Demo GitHub Repo:**

<https://github.com/atkuzmanov/openapi-demo-2022>

Disclosure: I bear no responsibility for the videos and code, or maintaining them, they come with no guarantees, warranties etc. use at your own risk.





CLASH OF THE TITANS - SPRING REST AND JAX-RS

CLASH OF THE TITANS: OpenAPI + Spring REST + JAX-RS



- So far, so good...
- Then you start adding OpenAPI in the real world, in a system which has gone through a few life-cycles and you encounter clash of the titans...
- A Java service application using both Spring REST and JAX-RS in the same code base...
- And you begin to try and make it work with OpenAPI for both...



Real-world Spring REST and JAX-RS clash of the titans

```
<plugin>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-maven-plugin</artifactId>
  <configuration>
    <mainClass>org.openapi.tools.OpenAPI2SpringBoot</mainClass>
    <executable>true</executable>
    <jvmArguments>-Dspring.application.admin.enabled=true</jvmArguments>
  </configuration>
  <executions>
    <execution>
      <id>pre-integration-test</id>
      <goals>
        <goal>start</goal>
      </goals>
    </execution>
    <execution>
      <id>post-integration-test</id>
      <goals>
        <goal>stop</goal>
      </goals>
    </execution>
  </executions>
  <execution>
    <id>repackage</id>
    <goals>
      <goal>repackage</goal>
    </goals>
  </execution>
</executions>
</plugin>
<dependency>
  <groupId>io.springfox</groupId>
  <artifactId>springfox-swagger2</artifactId>
  <version>${springfox-version}</version>
  <exclusions>
    <exclusion>
      <groupId>io.swagger</groupId>
      <artifactId>swagger-annotations</artifactId>
    </exclusion>
    <exclusion>
      <groupId>io.swagger</groupId>
      <artifactId>swagger-models</artifactId>
    </exclusion>
  </exclusions>
</dependency>
<dependency>
  <groupId>io.springfox</groupId>
  <artifactId>springfox-swagger-ui</artifactId>
  <version>${springfox-version}</version>
</dependency>
<dependency>
  <groupId>org.openapi.tools</groupId>
  <artifactId>jackson-databind-nullable</artifactId>
  <version>0.2.1</version>
</dependency>
<dependency>
  <groupId>javax.validation</groupId>
  <artifactId>validation-api</artifactId>
</dependency>
<dependency>
  <groupId>com.fasterxml.jackson.core</groupId>
  <artifactId>jackson-databind</artifactId>
  <version>2.11.4</version>
</dependency>
</dependencies>
</plugin>
```

```
<!-- START: OPENAPI 3.0 - swagger-request-validator -->
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-core</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-pact</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-wiremock</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-restassured</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-mockmvc</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>com.atlassian.oai</groupId>
  <artifactId>swagger-request-validator-spring-web-client</artifactId>
  <version>${atlassian.swagger-request-validator.version}</version>
</dependency>
<dependency>
  <groupId>au.com.dius</groupId>
  <artifactId>pact-jvm-consumer-junit_2.12</artifactId>
  <scope>test</scope>
  <version>3.6.15</version>
</dependency>
<dependency>
  <groupId>au.com.dius</groupId>
  <artifactId>pact-jvm-consumer_2.12</artifactId>
  <version>3.6.15</version>
</dependency>
<dependency>
  <groupId>au.com.dius</groupId>
  <artifactId>pact-jvm-provider_2.12</artifactId>
  <version>3.6.15</version>
</dependency>
<!-- END: OPENAPI 3.0 - swagger-request-validator -->
```

```
<!-- START: JAX-RS Jersey dependencies. -->
<dependency>
  <groupId>org.glassfish.jersey.containers</groupId>
  <artifactId>jersey-container-servlet-core</artifactId>
  <version>3.0.0</version>
</dependency>
<dependency>
  <groupId>org.glassfish.jersey.inject</groupId>
  <artifactId>jersey-hk2</artifactId>
  <version>3.0.0</version>
</dependency>
<dependency>
  <groupId>io.swagger.core.v3</groupId>
  <artifactId>swagger-jaxrs2</artifactId>
  <version>2.2.9</version>
</dependency>
<dependency>
  <groupId>io.swagger.core.v3</groupId>
  <artifactId>swagger-jaxrs2-servlet-initializer-v2</artifactId>
  <version>2.2.9</version>
</dependency>
<dependency>
  <groupId>javax.ws.rs</groupId>
  <artifactId>javax.ws.rs-api</artifactId>
  <version>2.1</version>
</dependency>
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>3.1.0</version>
</dependency>
<dependency>
  <groupId>org.glassfish.jersey.test-framework</groupId>
  <artifactId>jersey-test-framework-core</artifactId>
  <version>2.33</version>
  <scope>test</scope>
</dependency>
<dependency>
  <groupId>io.swagger</groupId>
  <artifactId>swagger-annotations</artifactId>
  <version>1.5.15</version>
</dependency>
<!-- END: JAX-RS Jersey dependencies. -->
```

OPENAPI + SPRING REST + JAXRS

pom.xml

Dependencies and plugins.

/src/main/resources/application.yml

Springdoc/Springfox dependency configurations.

/src/main/resources/openapi-configuration.yaml

Config for spec generation.

/src/main/java/com/example/config/JerseyConfig.java

Registering OpenApi resources for spec generation

/src/main/java/com/kaufland/property/directory/config/SwaggerConfig.java

Swagger configuration.

/src/main/java/com/kaufland/property/directory/controller/IndexController.java

A controller for the Swagger ui.

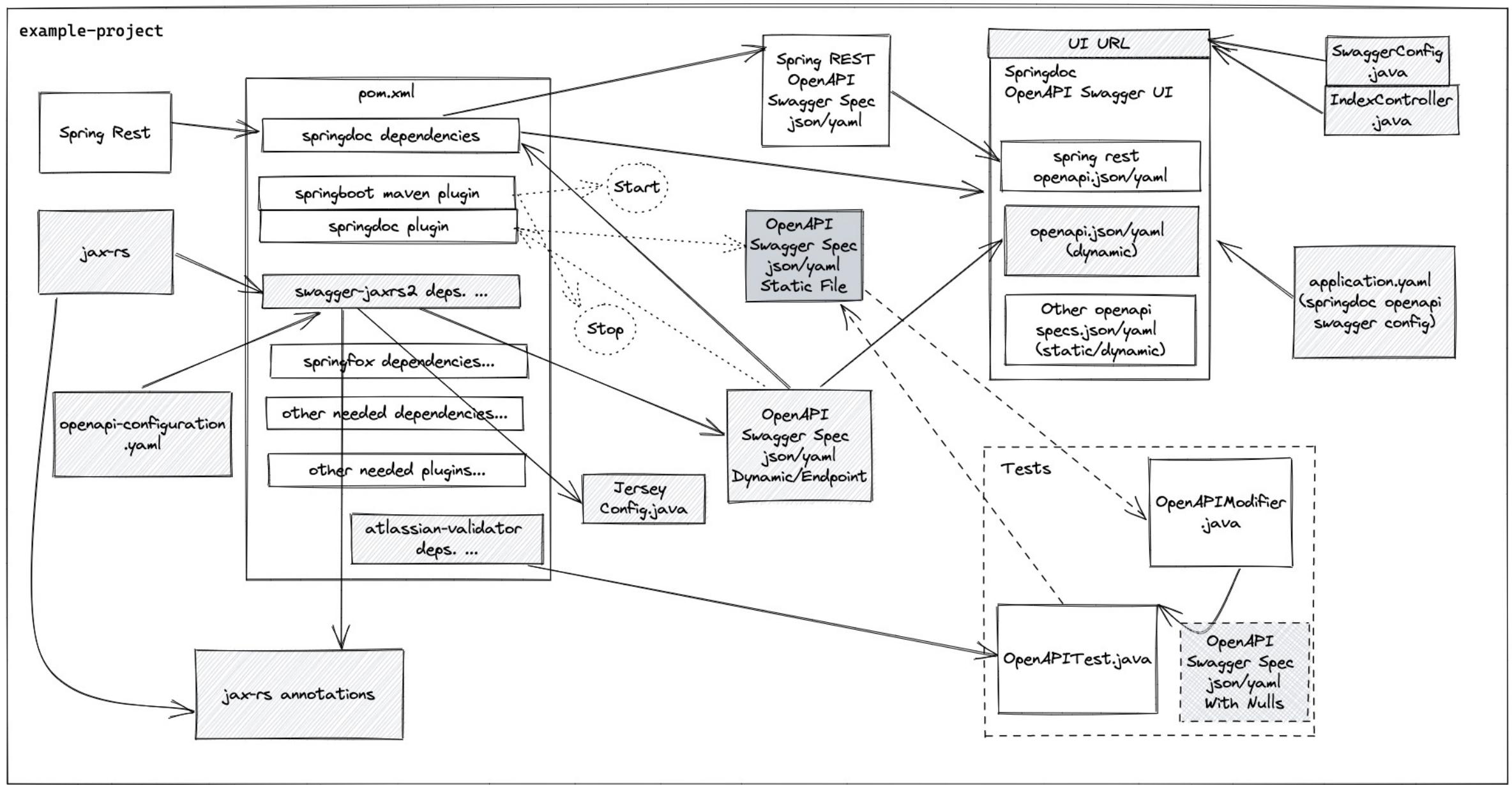
/src/test/java/com/example/service/OpenApiContractValidationTests.java

Tests using Atlassian's OpenApiValidationFilter to validate the code against an OpenAPI spec file.

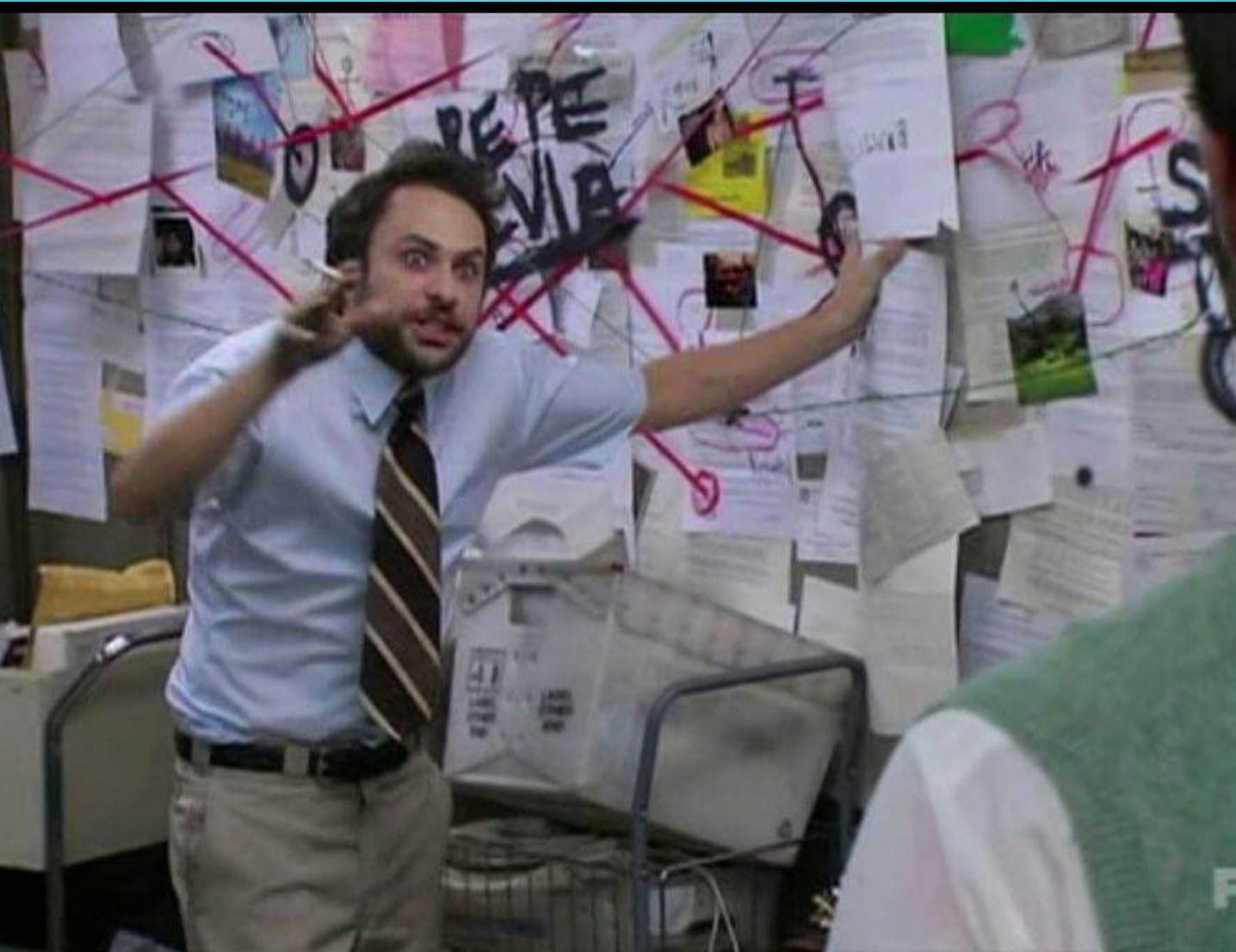
/src/test/java/com/example/OpenAPIModifier.java

Custom OpenAPIModifer class to modify our OpenAPI spec file before the tests run.

OPENAPI + SPRING REST + JAXRS



OPENAPI + SPRING REST + JAXRS





DEMO PART 2



DEMO PART 2 TODO

- Look at a code-to-spec scenario.
- Try to generate OpenAPI spec files from a Java application using both Spring REST and JAX-RS in the same code base.
- Highlight some of the difficulties encountered.
- Draw conclusions and lessons learned.



{DEMO}

{DEMO RECAP}

OpenAPI + Spring REST + JAX-RS Demo links:

- *Demo 2 Video Part 1:*

<https://www.youtube.com/watch?v=hB93Apbj5GQ>

<https://youtu.be/hB93Apbj5GQ>

- *Demo 2 Video Part 1:*

https://www.youtube.com/watch?v=go_9HquFdac

https://youtu.be/go_9HquFdac

- Demo 2 GitHub Repo:

<https://github.com/atkuzmanov/openapi-demo-2023-publ>

Disclosure: I bear no responsibility for the videos and code, or maintaining them, they come with no guarantees, warranties etc. use at your own risk.



CONCLUSIONS



CONCLUSIONS

In the current situation of increased remote working, having a common, simple medium of communication such as an OpenAPI spec document is indispensable in meeting your requirements when building APIs.

Things get even more exciting when this so-called document can:

- Serve as a contract
- Generate clients, servers, and documentation
- Put your developers and client first
- Have Automatic, Fail-fast, API Validation tests
- Is Cross-Platform and Language Agnostic
- Is both human and machine readable in YAML or JSON
- Speeds up development
- Reduces costs

Things we have to keep in mind:

- Even when you start with API-first writing a spec file first before writing any code, and then you write or generate the code, you eventually end up generating the spec from the code. This happens because over time when the code base matures it grows and evolves in different ways and often manipulating the spec manually becomes difficult, tedious and time-consuming. Moving the code-to-spec generation is the natural progression of an API First project in its lifecycle.
- The OpenAPI tools are getting better, but still need more development at least when it comes to Java.
- They work for simple use cases, but become unusable for more complex scenarios.
- A lot of them have vulnerabilities or compatibility issues due to lack of maintenance.
- Most of them are open source, so any contributions from the community are welcome for the greater good.
- Try not to use Spring REST and JAX-RS in the same codebase...



CLOSING THOUGHTS

CLOSING THOUGHTS

I would like to leave you with some lyrics from the song
"Everything Counts" by *Depeche Mode*:

"The handshake seals the contract

From the contract there's no turning back

...

The holiday was fun-packed

The contract, still intact



References:

<https://www.depechemode.de/lyrics/everything-counts-2/>
<https://www.youtube.com/watch?v=1t-gK-9Elq4>
https://en.wikipedia.org/wiki/Everything_Counts

“You Give REST a Bad Name”

A parody of Jon Bon Jovi's 'You Give Love a Bad Name'- about hypermedia APIs by Dylan Beattie.

"... Your API is a hall of shame..."

"... You give REST a bad name..."

"... you say it works in a restful way, then your errors come back as 200 OK..."



"... you promised me JSON, then sent XML..."



<https://www.youtube.com/watch?v=nSKp2StlS6s>

<https://youtu.be/nSKp2StlS6s?t=31>



References:

- <https://www.youtube.com/watch?v=nSKp2StlS6s>
- <https://www.youtube.com/channel/UCjJjavV8vOmu49a3vxPaWtQ>
- <https://dylanbeattie.net/>
- <https://twitter.com/dylanbeattie>
- <https://www.youtube.com/watch?v=KrZHPOeOxQQ>
- https://en.wikipedia.org/wiki/You_Give_Love_a_Bad_Name
- https://en.wikipedia.org/wiki/Bon_Jovi



REFERENCES

REFERENCES



- <https://swagger.io/>
- <https://swagger.io/specification/>
- <https://swagger.io/resources/articles/adopting-an-api-first-approach/>
- <https://swagger.io/docs/specification/about/>
- <https://swagger.io/blog/api-design/design-first-or-code-first-api-development/>
- <https://openapi-generator.tech/>
- <https://docs.pact.io/faq/convinceme/>
- https://pactflow.io/blog/why-contract-testing/?utm_source=ossdocs&utm_campaign=convince_me_what_is
- <https://www.atlassian.com/blog/technology/spec-first-api-development>
- <https://springframework.guru/defining-spring-cloud-contracts-in-open-api/>
- <https://springframework.guru/using-swagger-request-validator-to-validate-spring-cloud-contracts/>
- <https://springframework.guru/using-spring-cloud-contract-for-consumer-driven-contracts/>
- <https://openpracticelibrary.com/practice/contract-first-development/>
- https://link.springer.com/chapter/10.1007/978-3-540-71289-3_2
- <https://dzone.com/articles/an-api-first-development-approach-1>
- <https://medium.com/better-practices/api-first-software-development-for-modern-organizations-fdbfba9a66d3>
- <https://xkcd.com/1172/>
- <https://openclipart.org/detail/193532/contract>
- <https://apievangelist.com/2019/07/15/what-is-an-api-contract/>
- <https://apisyouwonthate.com/blog/api-design-first-vs-code-first>
- <https://www.visual-paradigm.com/guide/development/code-first-vs-design-first>
- <https://smartbear.com/blog/embracing-an-api-design-first-approach/>
- <https://www.depechemode.de/lyrics/everything-counts-2/>
- <https://www.youtube.com/watch?v=1t-gK-9Elq4>
- https://en.wikipedia.org/wiki/Everything_Counts
- <https://github.com/npm/logos/blob/master/npm%20logo/npm-logo-red.png>
- <https://www.npmjs.com/package/express-openapi-validate>
- <https://www.npmjs.com/package/express-openapi-validator>
- <https://de.wikipedia.org/wiki/Datei:Java-Logo.svg>
- https://en.wikipedia.org/wiki/TypeScript#/media/File:TypeScript_logo_2020.svg
- <https://commons.wikimedia.org/wiki/File:JavaScript-logo.png>
- <https://spring.io/trademarks>
- <https://angular.io/presskit>
- <https://twitter.com/SmartBear/status/914901646848679937>
- <https://www.teepublic.com/de/t-shirt/12045954-turn-ipa-into-api-coding>
- <https://openclipart.org/>
- <http://clipart-library.com/>
- <https://openclipart.org/detail/104323/integrated-development-environment>
- https://en.wikipedia.org/wiki/Schwarz_Gruppe
- <https://www.bbc.com/>
- <https://en.wikipedia.org/wiki/BBC>
- <https://www.twilio.com/company/brand>
- <http://clipart-library.com/clipart/377169.htm>
- <https://www.docker.com/company/newsroom/media-resources>
- <https://www.mongodb.com/brand-resources>
- http://clipart-library.com/clip-art/214-2149139_engineer-clipart-construction-plan-blue-safety-icon-png.htm
- <http://clipart-library.com/clip-art/light-bulb-clipart-transparent-12.htm>
- <https://openclipart.org/detail/237988/file>
- <https://openclipart.org/detail/188833/document-pictogram>
- <https://openclipart.org/detail/320714/man-in-business-suit>
- <https://pixabay.com/illustrations/group-discussion-human-personal-1962592/>

REFERENCES



- https://www.openapi.org/wp-content/uploads/sites/3/2016/10/OpenAPI_Pantone.png
- <https://openapi-generator.tech/img/mono-logo.svg>
- <http://clipart-library.com/clipart/lego-clipart-57.htm>
- <https://openclipart.org/detail/237996/package>
- <https://openclipart.org/detail/188650/search-ideogram>
- <https://openclipart.org/detail/190964/design>
- [https://en.wikipedia.org/wiki/Swagger_\(software\)#/media/File:Swagger-logo.png](https://en.wikipedia.org/wiki/Swagger_(software)#/media/File:Swagger-logo.png)
- http://clipart-library.com/clipart/agreement-cliparts_14.htm
- <https://openclipart.org/detail/250266/cruz>
- <https://openclipart.org/detail/194308/dev-minus>
- <https://openclipart.org/detail/275842/sisyphus-overcoming-silhouette>
- <https://openclipart.org/detail/307282/shaking-hands-silhouette>
- <https://openclipart.org/detail/215760/simple-gear>
- <https://openclipart.org/detail/305632/beer-and-barley>
- <https://www.euronatur.org/en/what-we-do/endangered-species/wolf/wolves-in-europe/>
- <http://www.wolfmatters.org/wolf-facts.html>
- <https://www.nathab.com/blog/8-incredible-wolf-facts-everyone-should-know/>
- <https://www.treehugger.com/wolf-facts-5120321>
- <https://onekindplanet.org/animal/wolf-grey/>
- <https://www.factretriever.com/wolves-facts>
- <https://facts.net/nature/animals/wolf-facts/>
- <https://openclipart.org/detail/276597/brain-profile-line-art>
- http://clipart-library.com/clip-art/308-3080052_freetoedit-silhouette-art-night-sky-moon-disney-wolf.htm
- https://www.duunddastier.de/wp-content/uploads/2018/09/Wolf_Titelthema_c_Fotolia_DUDT-3-18_Beitragbild.jpg
- <https://www.animalsprints.com/p/491/gray-timber-wolf-pair-canis-lupus-bozeman-area-7992771.jpg>
- <https://i.pinimg.com/originals/3c/1c/3d/3c1c3d238f290b47d4fdada054c3d9616.jpg>
- <https://i.pinimg.com/736x/6d/9f/53/6d9f5334e36f1ee02c73660a8247d185.jpg>
- <https://howlingforjustice.files.wordpress.com/2011/09/howling-wolf-1.jpg?w=584>
- [https://www.treehugger.com/thmb/ACY907vXT6tiv9IR3GxkfyU-VQ=/2113x0/filters:no_upscale\(\):max_bytes\(150000\):strip_icc\(\):format\(webp\)/GettyImages-1049557440-213fbe0089654a8684b2e081a5991963.jpg](https://www.treehugger.com/thmb/ACY907vXT6tiv9IR3GxkfyU-VQ=/2113x0/filters:no_upscale():max_bytes(150000):strip_icc():format(webp)/GettyImages-1049557440-213fbe0089654a8684b2e081a5991963.jpg)
- <https://openclipart.org/detail/124147/wolf-3>
- <https://openclipart.org/detail/26795/siamese-cat>
- <https://openclipart.org/detail/313039/german-shepherd-sketch>
- <https://www.oreilly.com/library/view/head-first-java/0596009208/>
- <https://www.oreilly.com/library/view/head-first-design/0596007124/>
- <https://en.wikipedia.org/wiki/API>
- <https://www.mulesoft.com/resources/api/what-is-an-api>
- <https://blog.axway.com/amplify-products/api-management/what-is-an-api>
- <https://openclipart.org/detail/13273/server>
- <https://openclipart.org/detail/318251/multiple-screens>
- https://media-exp1.licdn.com/dms/image/C4D22AQG_QxsuoBiHuQ/feedshare-shrink_800/0/1636456936477?e=1639612800&v=beta&t=EKHU4qisLTvNkM94SnZBp8SJLXAcjmAG04SInP_2VqY
- <https://twitter.com/tompahoward/status/1458618578500874242/photo/1>
- <https://stackoverflow.com/questions/63834848/how-to-document-graphql-with-swagger-openapi>
- <https://github.com/graphql/graphql-playground>
- <https://github.com/IBM/openapi-to-graphql>
- <https://github.com/yarax/swagger-to-graphql>
- <https://loopback.io/openapi-to-graphql.html>
- <https://en.wikipedia.org/wiki/GraphQL>
- <https://www.akana.com/blog/api-first>
- <https://auth0.com/blog/the-business-value-of-api-first-design/>
- <https://stoplight.io/case-studies/wefox/>
- <https://me-qr.com/>
- <https://www.youtube.com/watch?v=nSKp2StIS6s>
- <https://www.youtube.com/channel/UCJjavV8vOmu49a3vxPaWtQ>
- <https://dylanbeattie.net/>
- <https://twitter.com/dylanbeattie>
- <https://www.youtube.com/watch?v=KrZHPOeOxQQ>
- https://en.wikipedia.org/wiki/You_Give_Love_a_Bad_Name
- https://en.wikipedia.org/wiki/Bon_Jovi
- <https://jprime.io/>
- <https://i.imgur.com/1arqp8.jpg?a467328>
- https://principles.schwarz/press-releases/one-direction/questions/api_first/
- <https://i.kym-cdn.com/photos/images/newsfeed/000/517/235/23e.jpg>



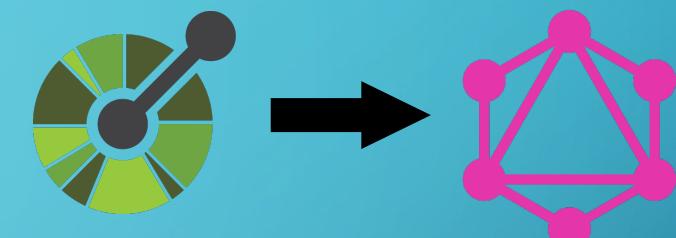
QUESTIONS



ICEBREAKER QUESTIONS

- *Can you use OpenAPI for GraphQL?*

No.



Tools for translation:

- <https://github.com/graphql/graphql-playground>
- <https://github.com/IBM/openapi-to-graphql>
- <https://github.com/yarax/swagger-to-graphql>
- <https://loopback.io/openapi-to-graphql.html>



QUESTIONS



CONTACTS

Schwarz Global Services
Bulgaria EOOD

FULL of prospects.

» A job like no other.

Office X, Floor 9, Schwarz IT,
bulevard "Cherni vrah" 51,
1407 Promishlena zona
Hladilnika, Sofia



<https://dev.bg/company/schwarz-it/>



<https://it.schwarz/en/international/bulgaria>



<https://www.jobs.bg/company/schwarz-it/>



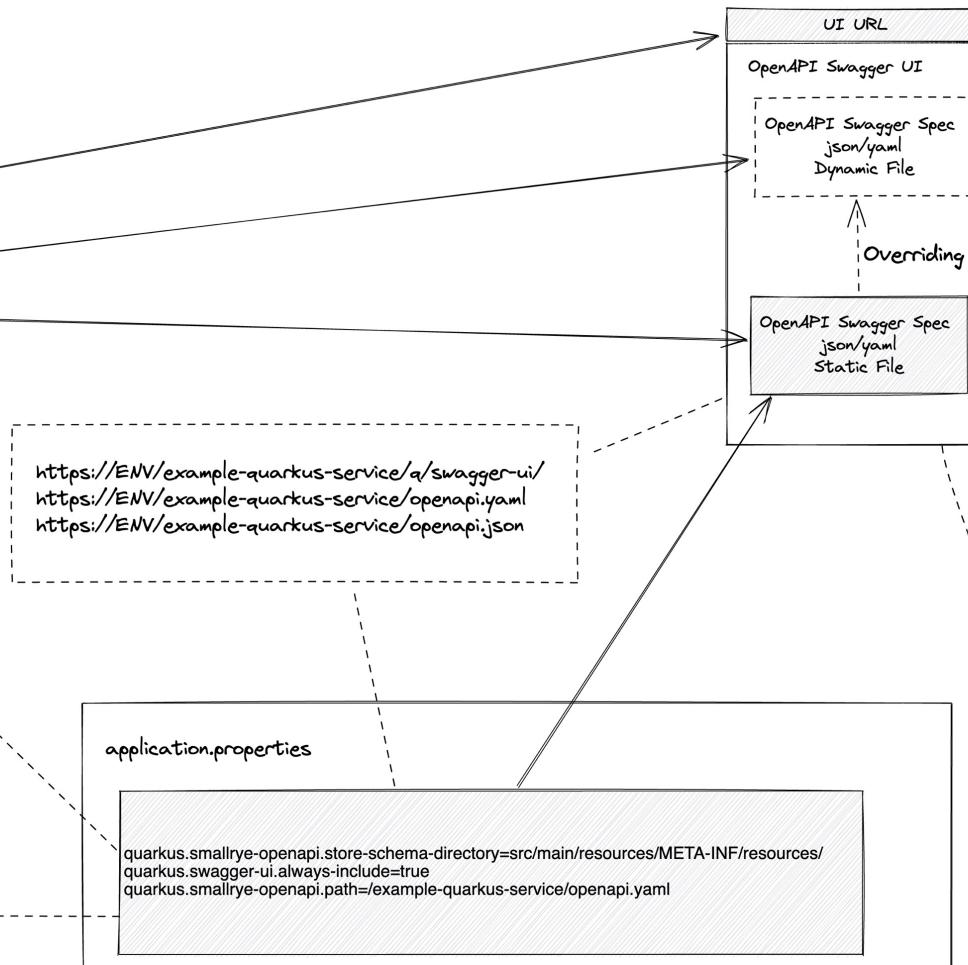
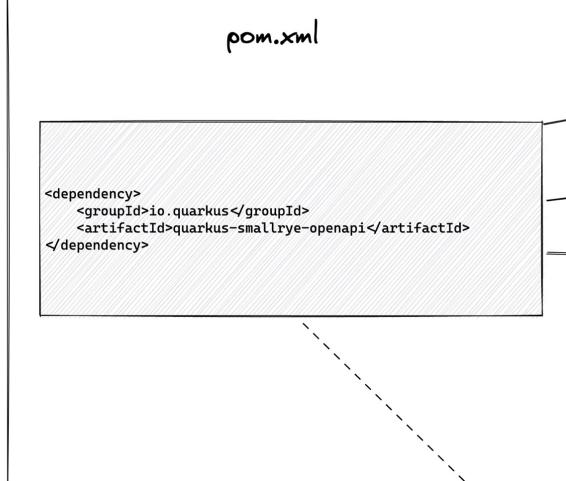
THANK YOU!



{BONUS}

OPENAPI + QUARKUS

example-quarkus-service



WOLF FACT



It's all in the howl

- Wolves do howl at night, and these soulful calls have nothing to do with the moon. Their penetrating, hauntingly beautiful howl is a primary communication tool, both between lone wolves and their pack, as well as between packs.



THANK YOU!

