
SPS COMMERCE
API
API DESIGN FIRST
SUCCEEDING WITH
API GOVERNANCE
APIS FROM DEVELOPERS FOR DEVELOPERS!
Sponsors
accenture
<dev>pro
Community Sponsor
SGI

TRAVIS GOSSELIN

DISTINGUISHED SOFTWARE ENGINEER

DEVELOPER EXPERIENCE

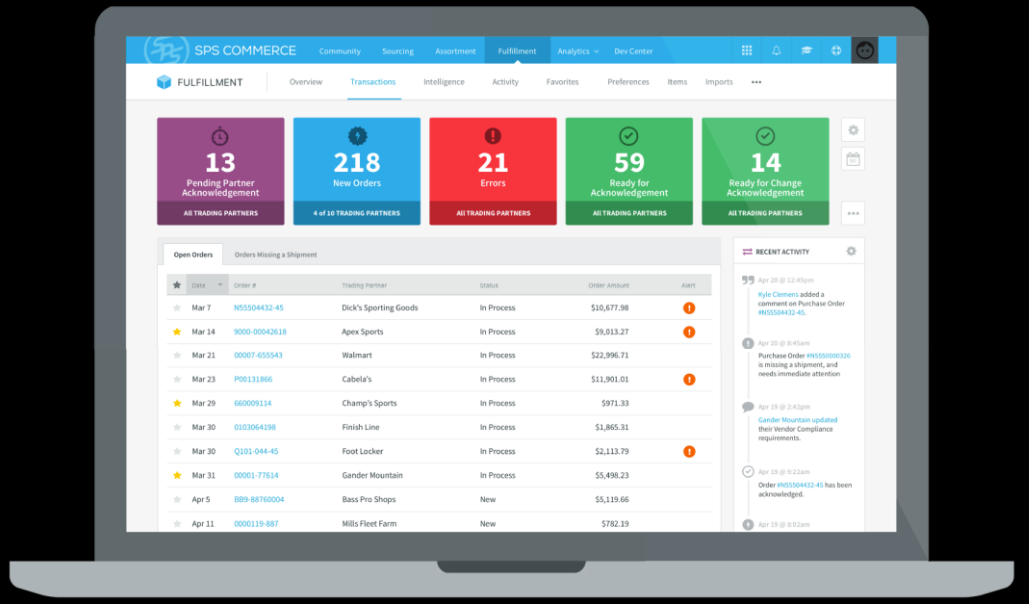
travisgosselin.com

linkedin.com/in/travisgosselin

@travisjgosselin

SPS COMMERCE

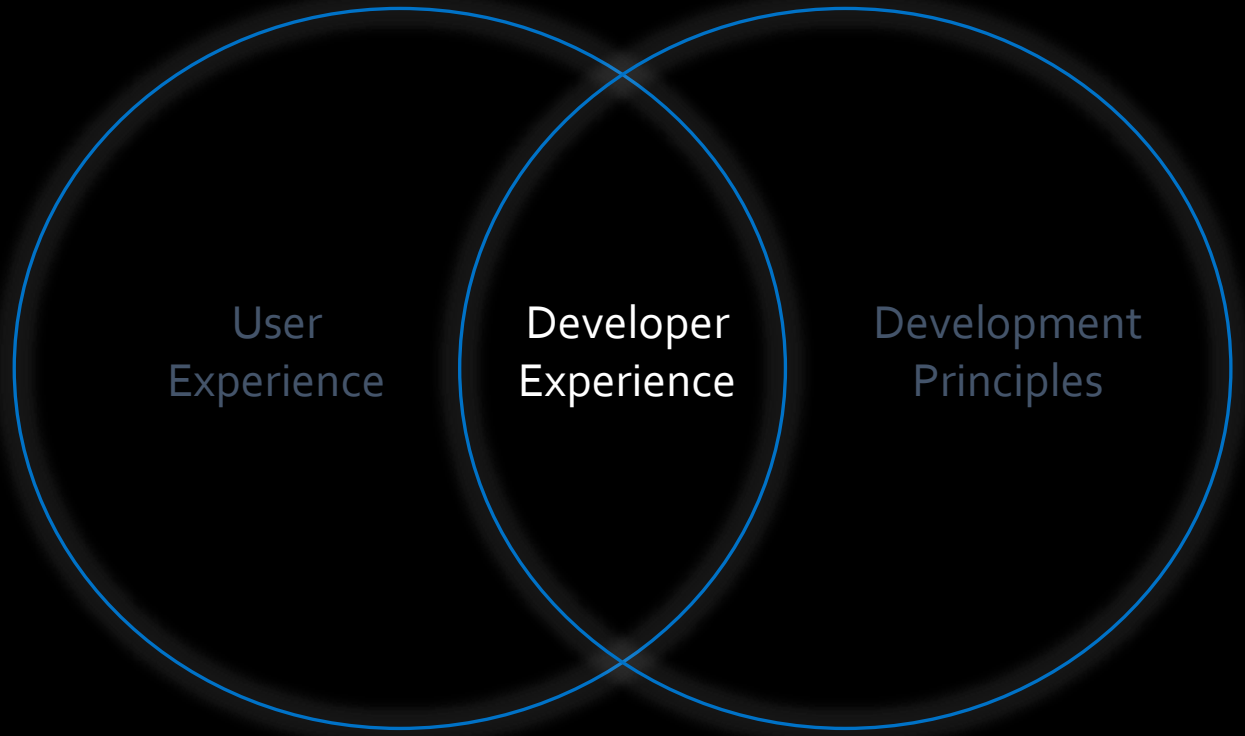
INFINITE RETAIL POWER™



> Developer Experience is the activity of studying, improving and optimizing how developers get their work done.
>
> theappslab.com (2017)

DEVELOPER EXPERIENCE

WHAT IS THAT…EXACTLY?



> Developers work in rainforests, not planned gardens.
>
> a16z.com

DEVELOPER EXPERIENCE == APIS

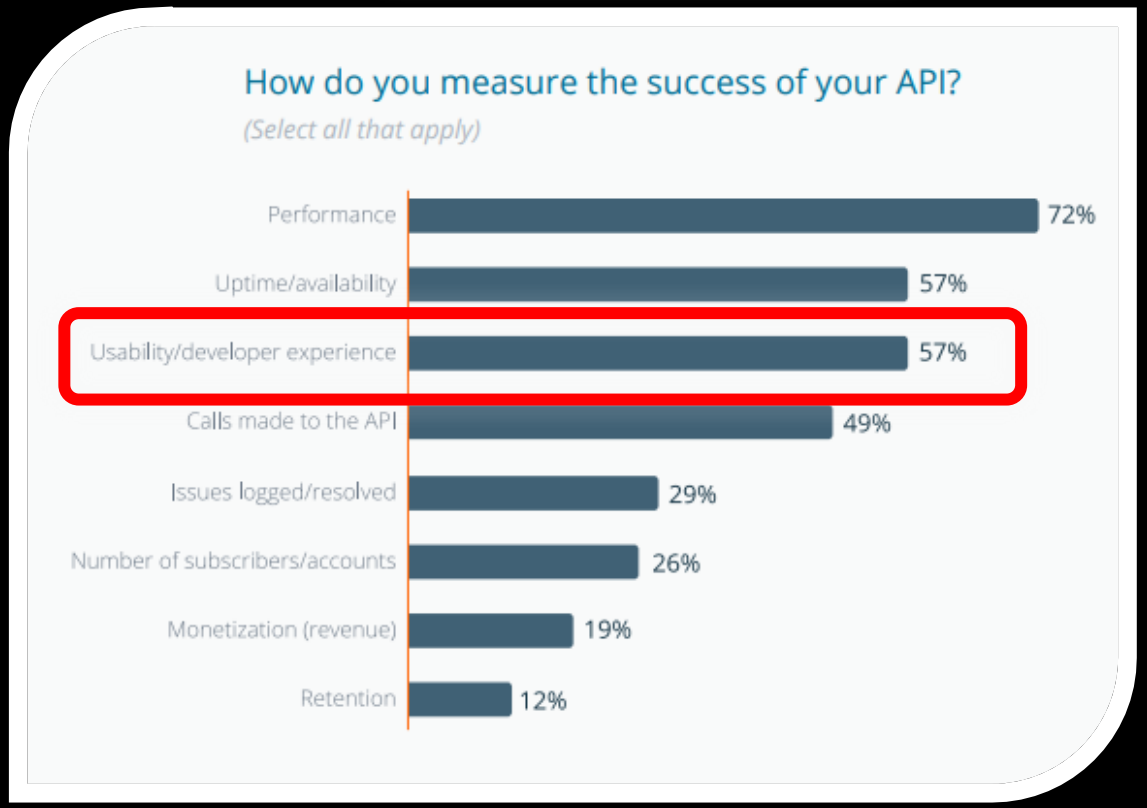
APIs are the Universal Language of Development

Over 90% of Developers Use APIs

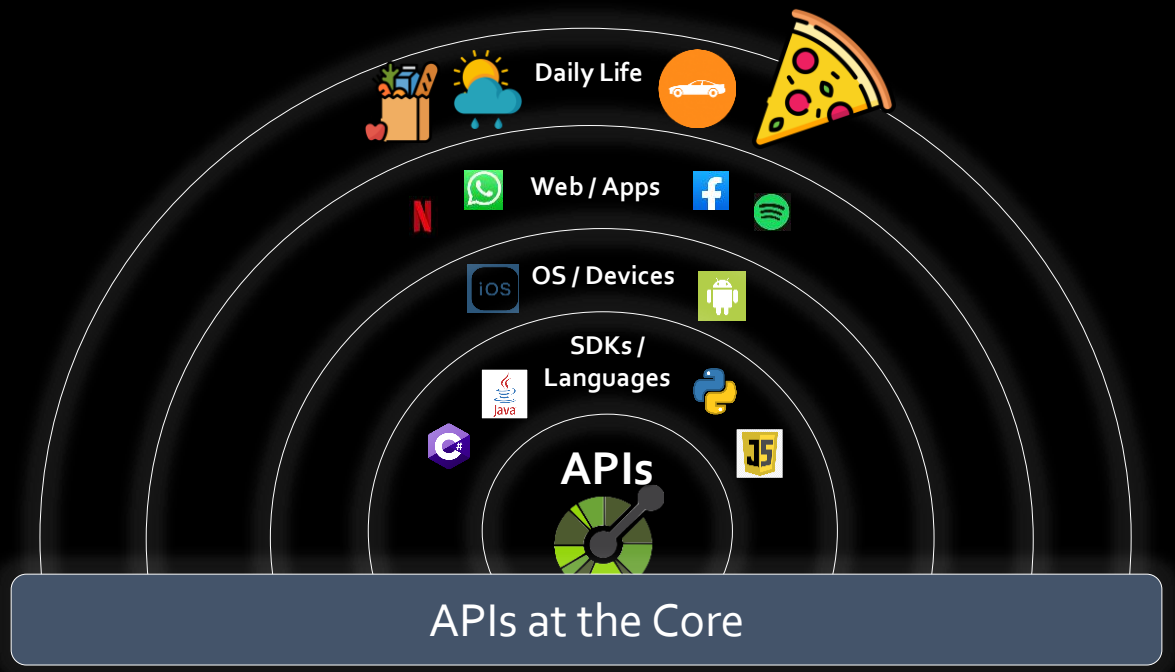
2020 NordicAPI

Developers Spend 30% of Their Time Coding APIs

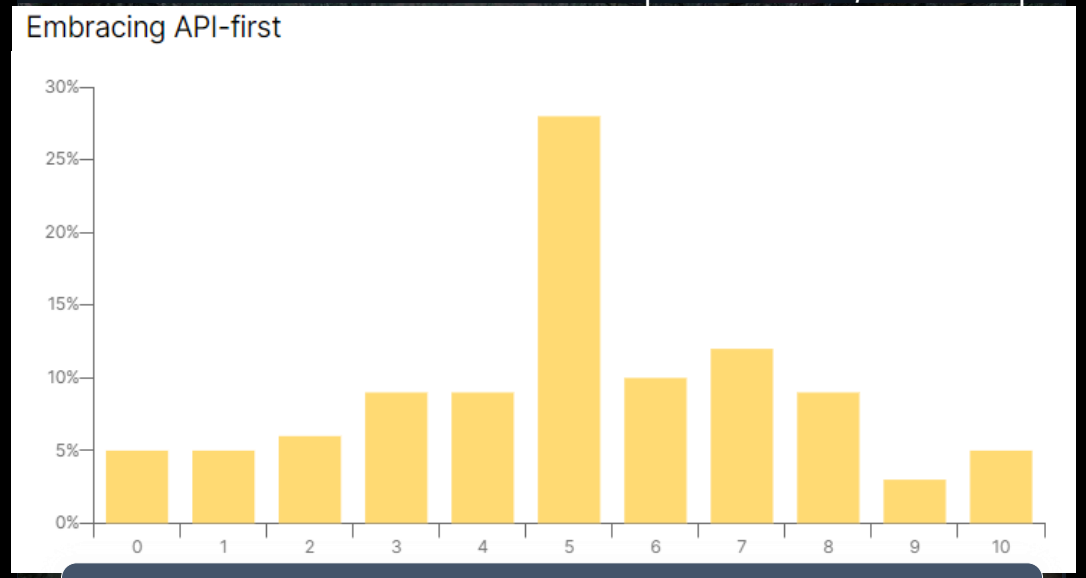
Second to Manual Debugging: 17%



SmartBear 2020 State of API



postman.com/state-of-api

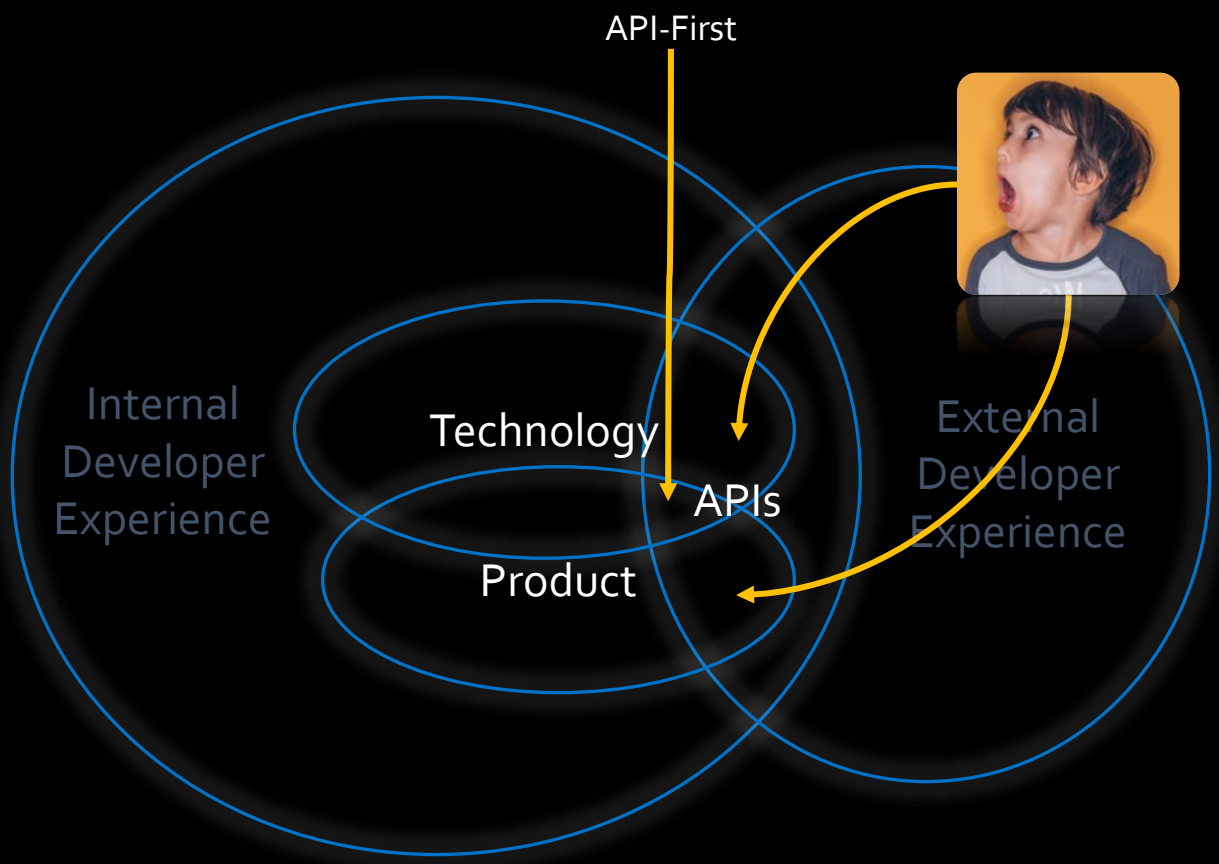


84% of API-First Developers Are Happier!

API-FIRST APPROACH

I BUILD APIS... SAME THING?

> An API-first approach means that [...] your APIs are treated as "first-class citizens." That everything about a project revolves around the idea that the end product will be consumed [...] by client applications.
>
> swagger.io



"When a design-first approach is not followed, the development process can become chaotic and disorganized. These projects often experience lapses and disconnects between the developers, security, governance, and documentation teams.

[...] It's like being brought in as a building inspector for a house that was built with no blueprints."

Stoplight

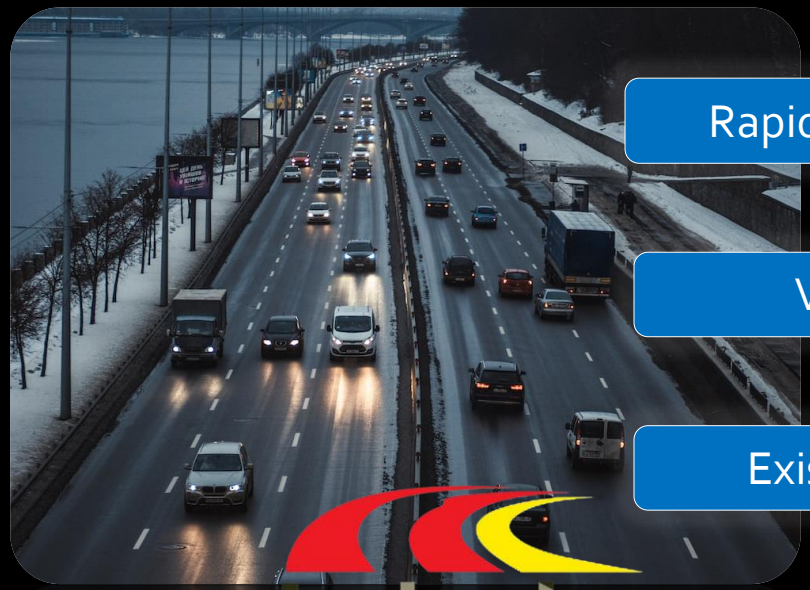
The API Roadmap from Stoplight (2022)

DESIGNING YOUR API

API GOVERNANCE

Natural Tension in the Enterprise

What we *feel* we want



Rapid Evolution ⟷ No Breaking Changes

Velocity ⟷ Developer Experience

Existing APIs ⟷ Guidelines / Consistency

What we *feel* we need

Mark Weitzel, Building APIs at Scale: Moving from API Governance to API Stewardship

API GOVERNANCE

Natural Tension in the Enterprise

What we *actually* need:

Modern Roundabout

- Specific design and traffic control features
  - Yield control of entering traffic
  - Priority given to circulating traffic
  - Appropriate curvature to ensure low travel speeds
- Designed to create free-flowing, low-speed environment
  - Increases intersection efficiency
  - Eliminates the need for storage lanes
- Safety is enhanced by the elimination of conflict points
  - Possibility for head on collisions drastically reduced
  - 90% reduction in deaths compared to signal controlled intersections

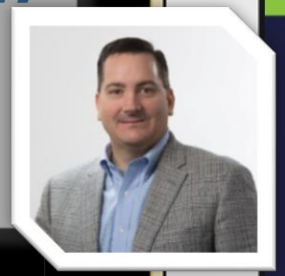
https://pw.lacounty.gov

API GOVERNANCE

MODERNIZING API GOVERNANCE THROUGH API STEWARDSHIP

https://www.youtube.com/watch?v=sboH7zyfEnA

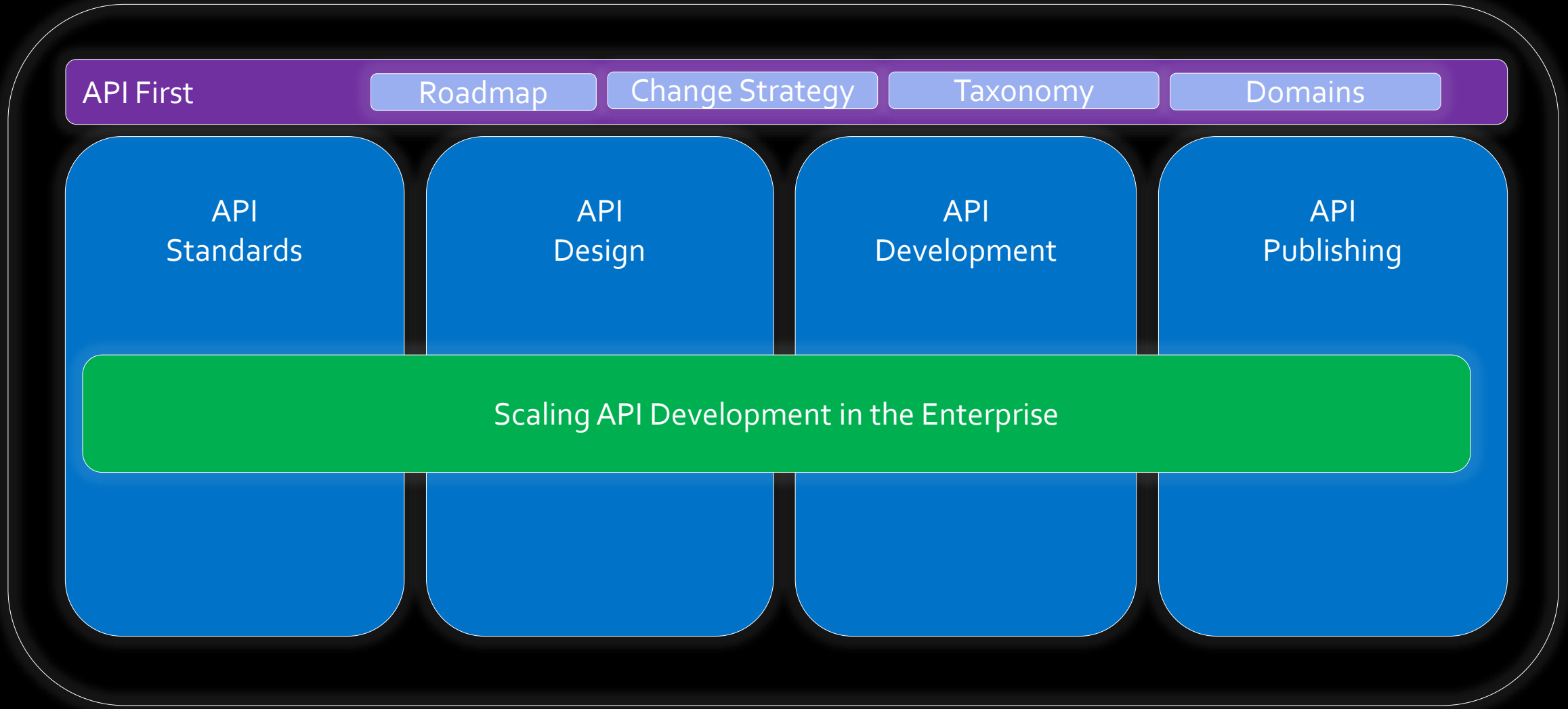
> **Stewardship**: The careful and responsible management of something entrusted to one's care.



Grow an orchard, don't inspect the fruit.

API DESIGN MEETS GOVERNANCE

Building API Governance into the API Lifecycle Components



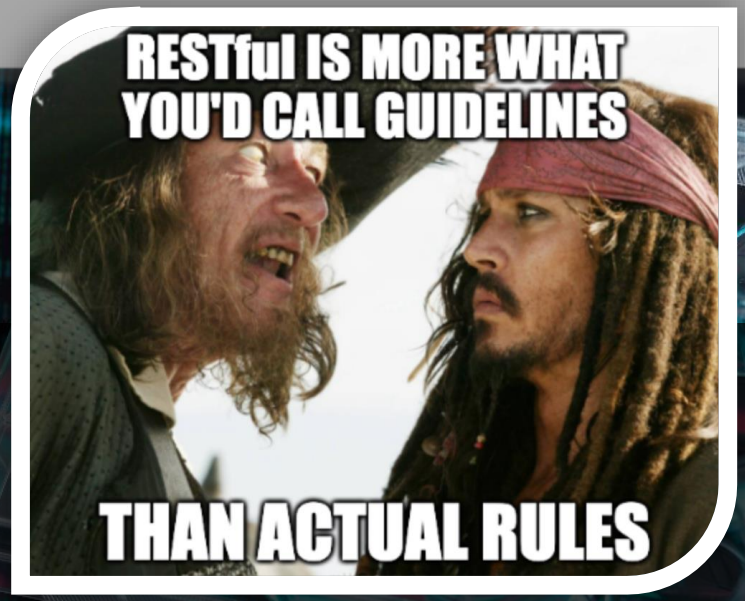
"

STANDARDS

: a level of quality or attainment.

"

Google.com



API STANDARDS

REST IS AN ARCHITECTURE, NOT A STANDARD!

- API Standards
- API Design
- API Development
- API Publishing

API Challenges & Future Growth

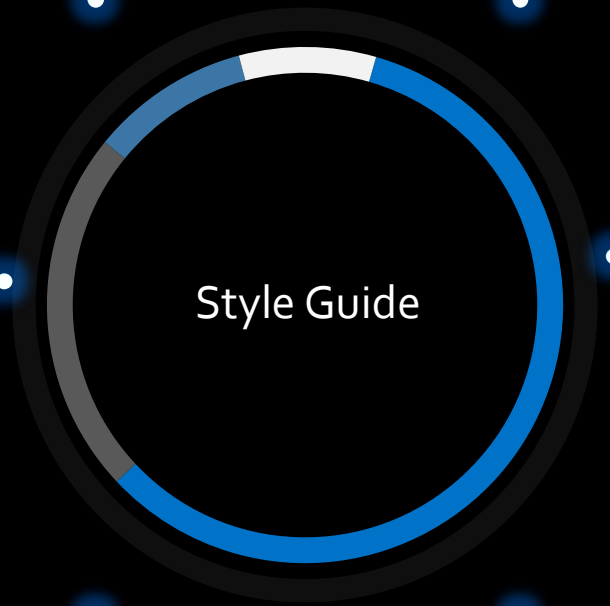
Standardization continues to be the top API technology challenge teams want to see solved in the coming years

Smart Bear 2020 State of API Report

API STYLE GUIDE

Design Guidelines | API Standards | Style Books

Style guides inform our decisions when designing API contracts & interfaces.



URL Structure

Resource Paths, Hierarchy, Actions, Query Parameters, Fragments

Collections

Root Elements, Pagination, Filtering and Sorting

Serialization

Casing, Types, Quantities, Intervals, Durations, etc

Request & Response

HTTP Status Codes, Headers, Verbs, Hypermedia

Errors

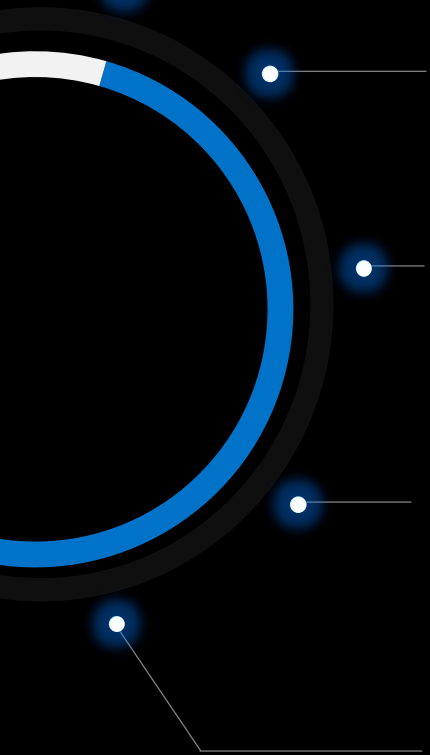
Standard Error Responses, Schema, and Required Fields (i.e. X-Request-ID).

Naming

General Practices, Reused Names, Standard Properties

API STYLE GUIDE

Best Practices



Less Is More

Keep it brief; not a book; not an implementation guide

Living Document

The guide will mature over time with real-world use cases.

Goldilocks Use Cases

Beneficial to not over-burden certain standards.

Align to Existing Standards

Use HTTP and RESTful primitives as intended; use other well-known RFCs!

Examples ALWAYS!

Inline; post; summary… examples for all! INCORRECT EXAMPLES!

- Bullet points
- Clear statements
- Avoiding statements suggesting implementation

Awareness of implementation implications will direct your style guide for quicker and wider adoption!

MUST: means that the definition is an absolute requirement of the specification.
SHOULD: means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications must be understood and carefully weighed before choosing a different course.

https://www.ietf.org/rfc/rfc2119.txt

HTTP/1.1 403 Forbidden PROBLEM DETAILS
Content-Type: application/problem+json

```
{
  "type": "https://example.com/out-of-credit",
  "title": "You do not have enough credit.",
  "detail": "Your current balance is 30, but that costs 50.",
  "instance": "/account/12345/msgs/abc",
}
```

RFC7807 REPLACED BY RFC9457

API STYLE GUIDE

Examples

api-standards/api-style-guide.md

github.com/paypal/api-

2474 lines (1779 sloc)

HTTP Meth

Data Resources

Business Capabiliti

Various business capabi
duplicated across APIs;
use-cases.

HTTP Methods

Most services will fall ea
acronym CRUD (Create,

HTTP Method	
GET	To
POST	To
PUT	To
DELETE	To
PATCH	To

api-guidelines/Guidelines.md

github.com/Microsoft/api

2159 lines (1641 sloc)

7.4 Supported meth

Operations MUST use the
methods are frequently
specification uses the ter

Below is a list of methods
resources using the meth

Method	
GET	Return the
PUT	Replace a
DELETE	Delete an
POST	Create a n
HEAD	Return me method M
PATCH	Apply a pa
OPTIONS	Get inforn

Table 1

7.4.1 POST

POST operations SHOULD

api-design-guide/README.md

github.com/CiscoDevNet/api-design-guide/blob/master/README.md

1141 lines (747 sloc) | 56.3 KB | Raw | Blame

3.6 HTTP Verbs

3.6.1 POST

3.6.1.1 A POST operati
request.

3.6.1.2 A POST operati
request.

3.6.1.3 A POST operati
within the collection e
by a unique service-ge
resource. This URL MU
response, as well as th

be regarded as an alternative forms

UST be regarded as an action resource

rded as a request to create a new resource
o such a request MUST be accompanied
int, and referring to the newly created
a `Location` header in the HTTP

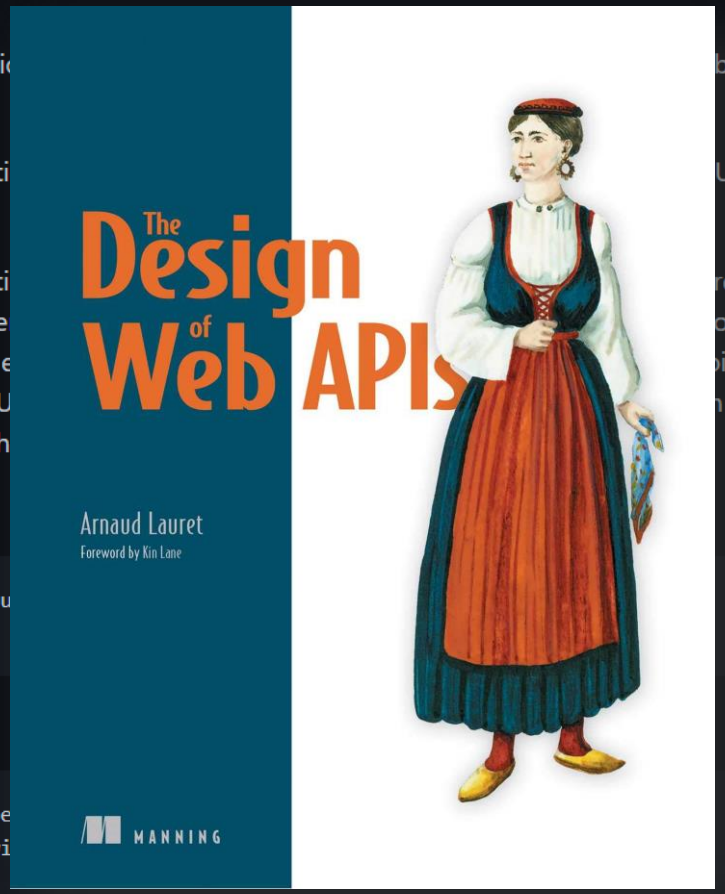
example request:

```
POST /files/v3/docu
...
```

example response:

```
HTTP/1.1 201 Create
Location: http://wi
...

{
    "url" : "http://widgets.example.com/files/v3/documents/35bd3e"
}
```



API Handyman: Arnaud Lauret
ApiStyleBook.com

SPS STYLE GUIDE

SPS COMMERCE REST API STANDARDS

https://github.com/spscommerce/sps-api-standards

cloud foundry

Consistency – understanding one resource informs interacts with the rest.

Discoverability – API responses guide users without the need for external documentation.

Simplicity – complex user workflows constructed from smaller parts.

Opinionatedness – one clear way to do something.

Tolerance – contracts and consumers are forgiving as possible without compromising security.

Automation – lean towards defendable standards, where not compromising

Experience – developer experience is king, and major concerns trump the rest.

API STYLE GUIDE

No One Really Cares About Your Guide!

- They WILL NOT be read it.
- Teams will agree with it and deviate.
- Engineers WILL ask questions to answers they could have just searched for or read (LMGTFY).
- Individuals WILL use interpretations that represent conflicts or contradictions "just to be that guy".
- Years later complaints will continue for decisions already made.

https://apihandyman.io/nobody-cares-about-api-design-guidelines/

https://spscommerce.github.io/sps-api-standards

API STYLE GUIDE

SPS COMMERCE

Automation

- Every error response (4xx/5xx) SHOULD use the Problem Details RFC 7807 Format

```
spectral lint api.yml --ruleset .spectral.yml
```

```
spectral lint api.yml --ruleset https://raw.githubusercontent.com/SPSCommerce/sps-api-standards/main/sps-api-standards.spectral.yml
```

```yaml
--- .spectral.yml
extends:
  - "spectral:oas“
  - your-other-rules.yml
rules:
  operation-description: off

problem-details-error-response:
  description: "Every error response SHOULD support RFC 7807
  severity: error
  given: $.paths.[*].responses[?(@property.match(/^(4|5)/))].content.*~
  then:
    function: enumeration
    functionOptions:
      values:
```

```
 1:1   warning  oas3-api-servers       OpenAPI "servers" must be present and non-empty array.
 2:6   warning  info-contact           Info object must have "contact" object.                          info
 2:6   warning  info-description       Info "description" must be present and non-empty string.         info
 7:9   warning  operation-description  Operation "description" must be present and non-empty string.  paths./v1/users.get
 7:9   warning  operation-operationId  Operation must have "operationId".                             paths./v1/users.get
 9:9   warning  operation-tag-defined  Operation tags must be defined in global tags.                 paths./v1/users.get.tags[0]
20:30    error  unknown-error-format   Every error response SHOULD support RFC 7807.                 paths./v1/users.get.responses[400].content.application
```

"I think most people just make the mistake that it should be simple to design simple things.

In reality, the effort required to design something is inversely proportional to the simplicity of the result."

Roy T. Fielding

DESIGN

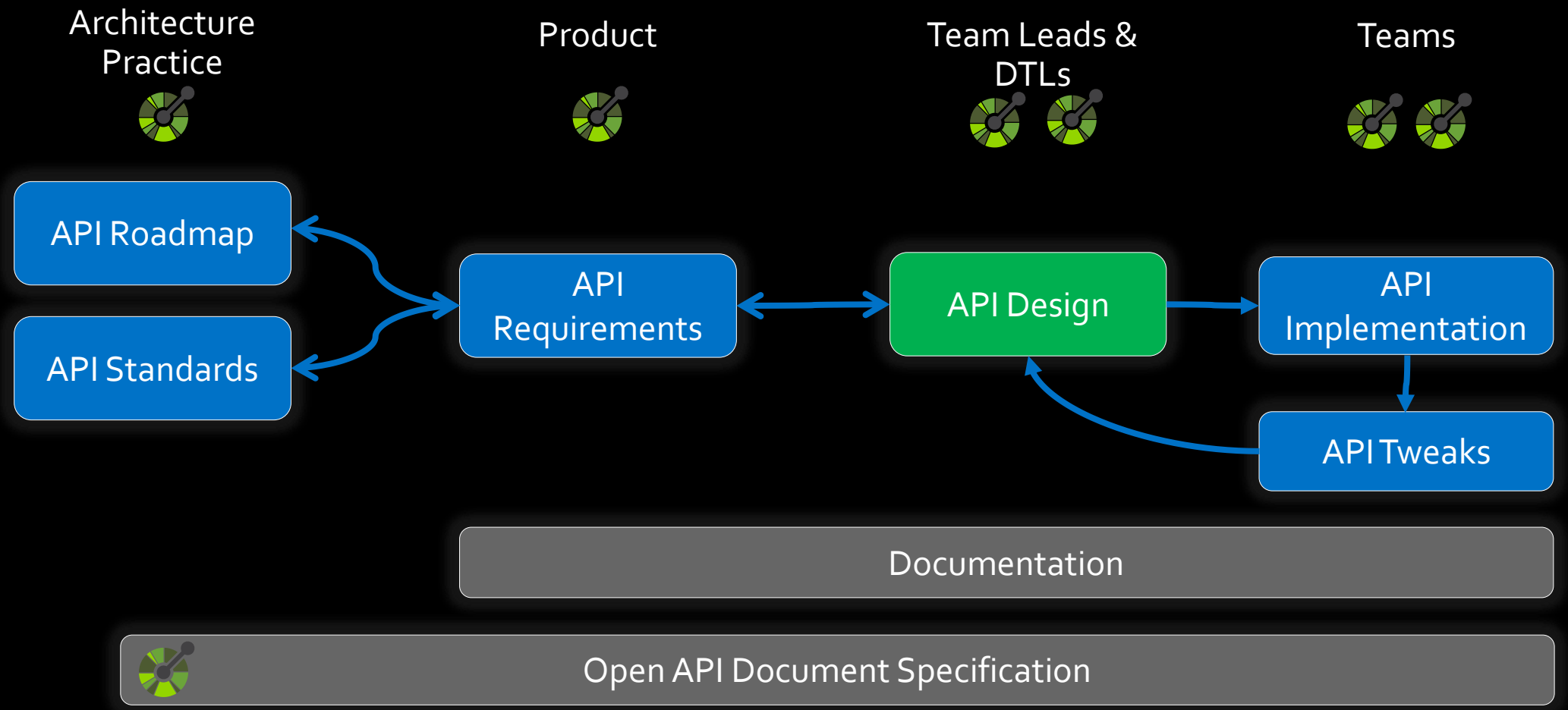
THIS IS WHY IT IS "API DESIGN FIRST"

- API Standards
- **API Design**
- API Development
- API Publishing

Process	People	Technology
Design Reviews and Workflow	Navigating Opinions	Tooling, Automation

DESIGN

Organizational Breakdown



DESIGN

Review

Collaboration

Helping People
Making it Better
They Are the Owner

Transparency

Everyone
Stakeholders
Product
Other Teams

Consistency

API Standards &
Models

- Resources
- Naming
- Collections

Strategy & Workflow

Chatty?

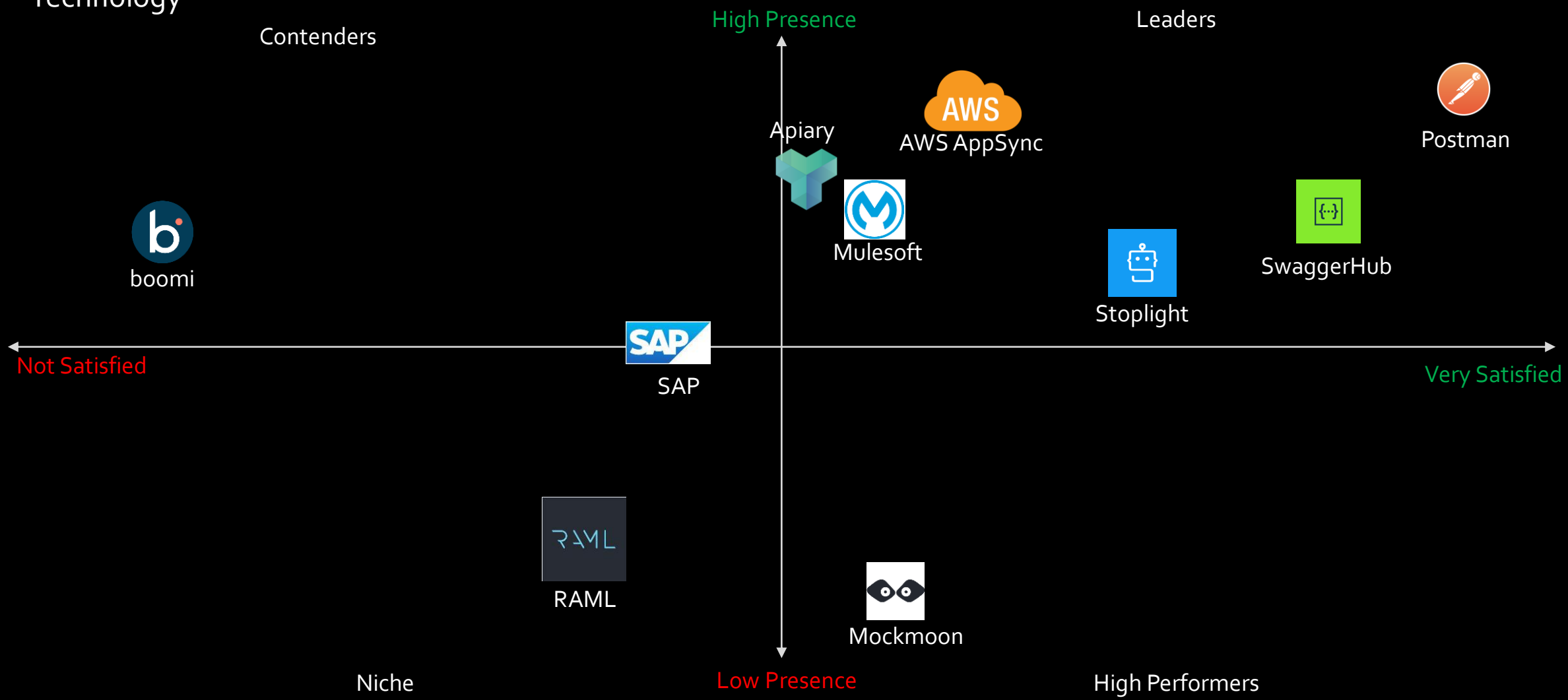
- Async / Bulk
- Resiliency
- Exceptions

Learning Opportunities

Everyone is Learning

DESIGN

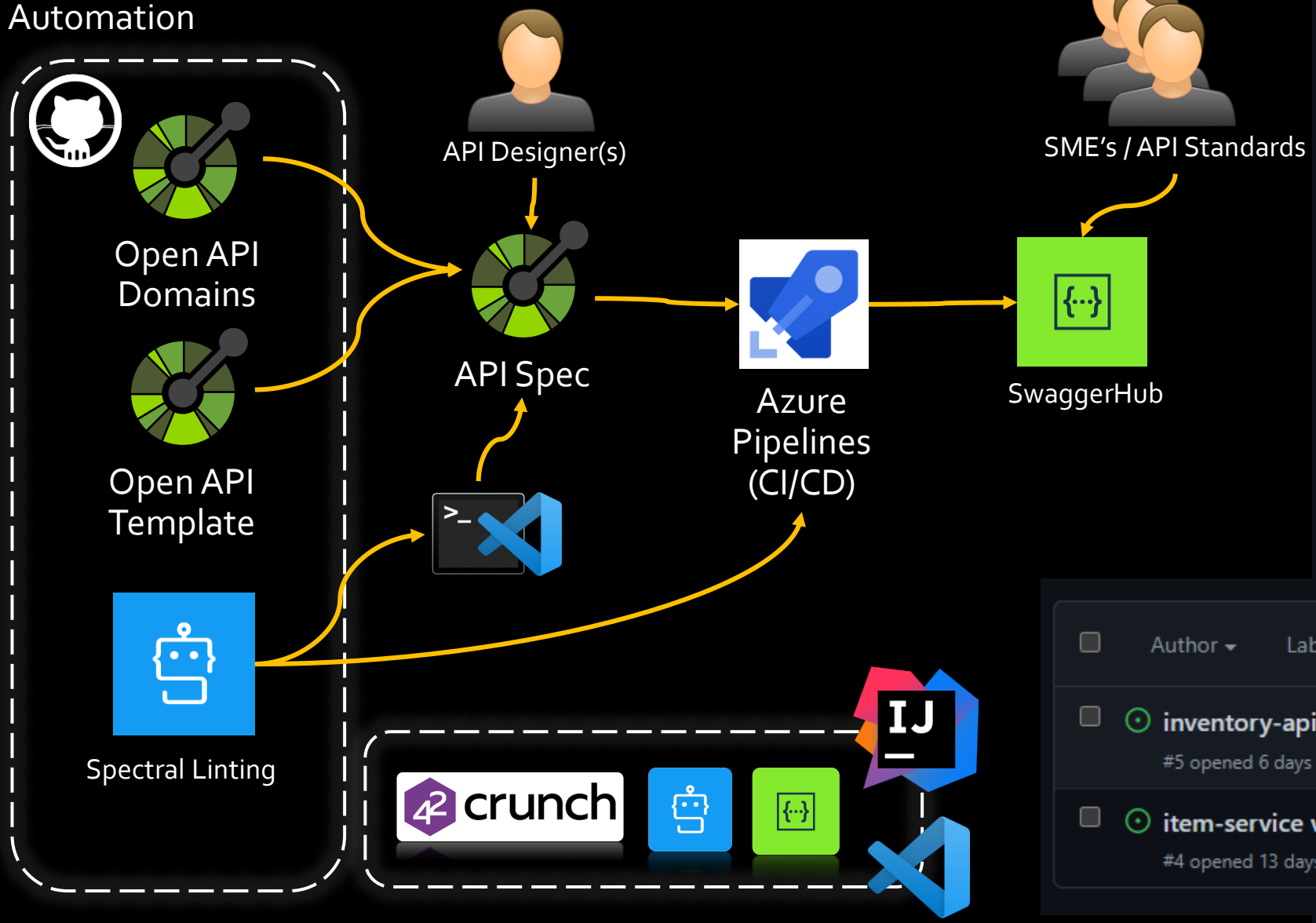
Technology

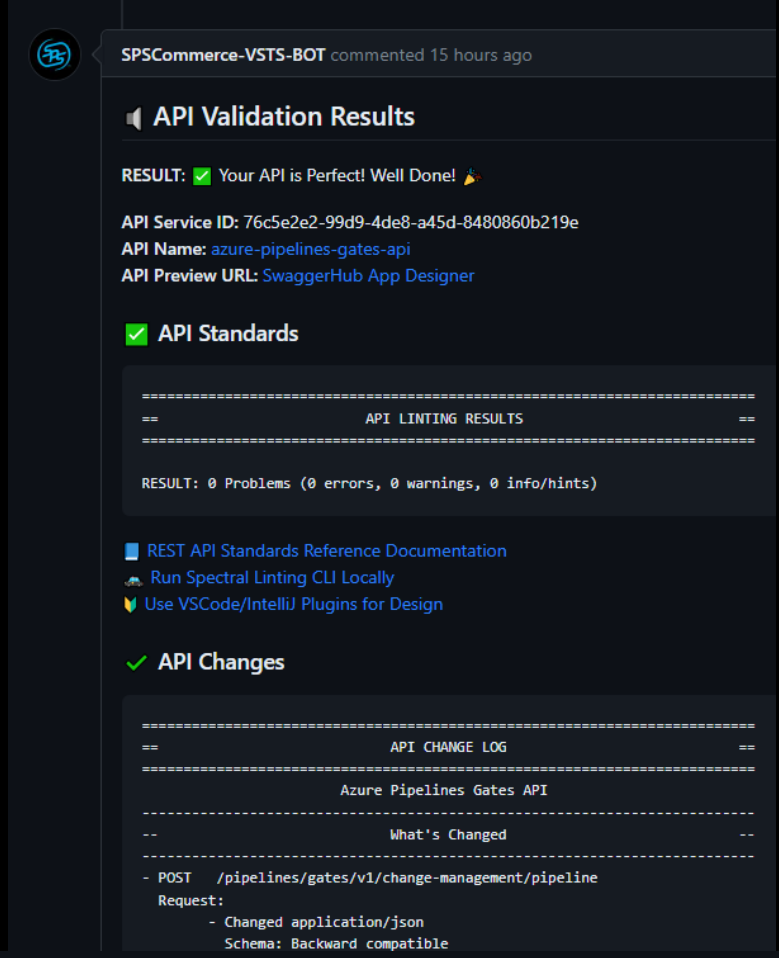


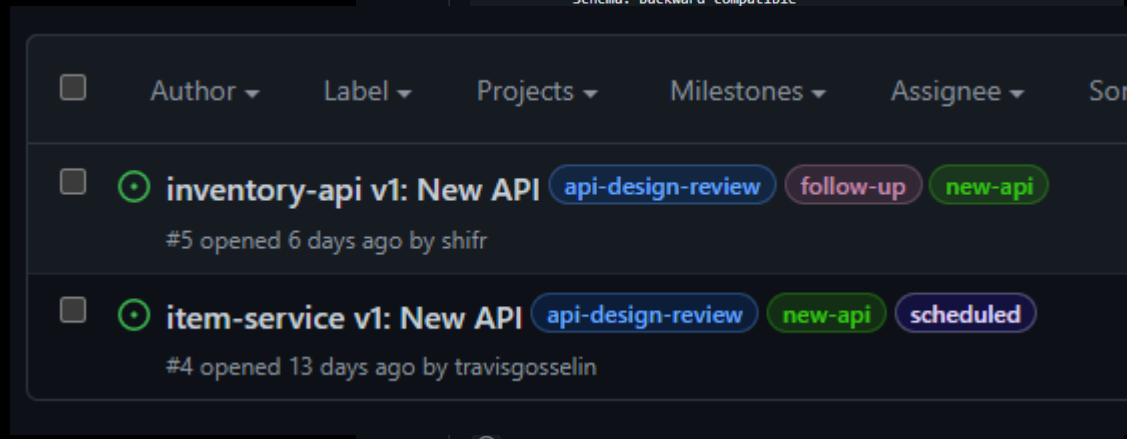
https://www.g2.com/categories/api-design

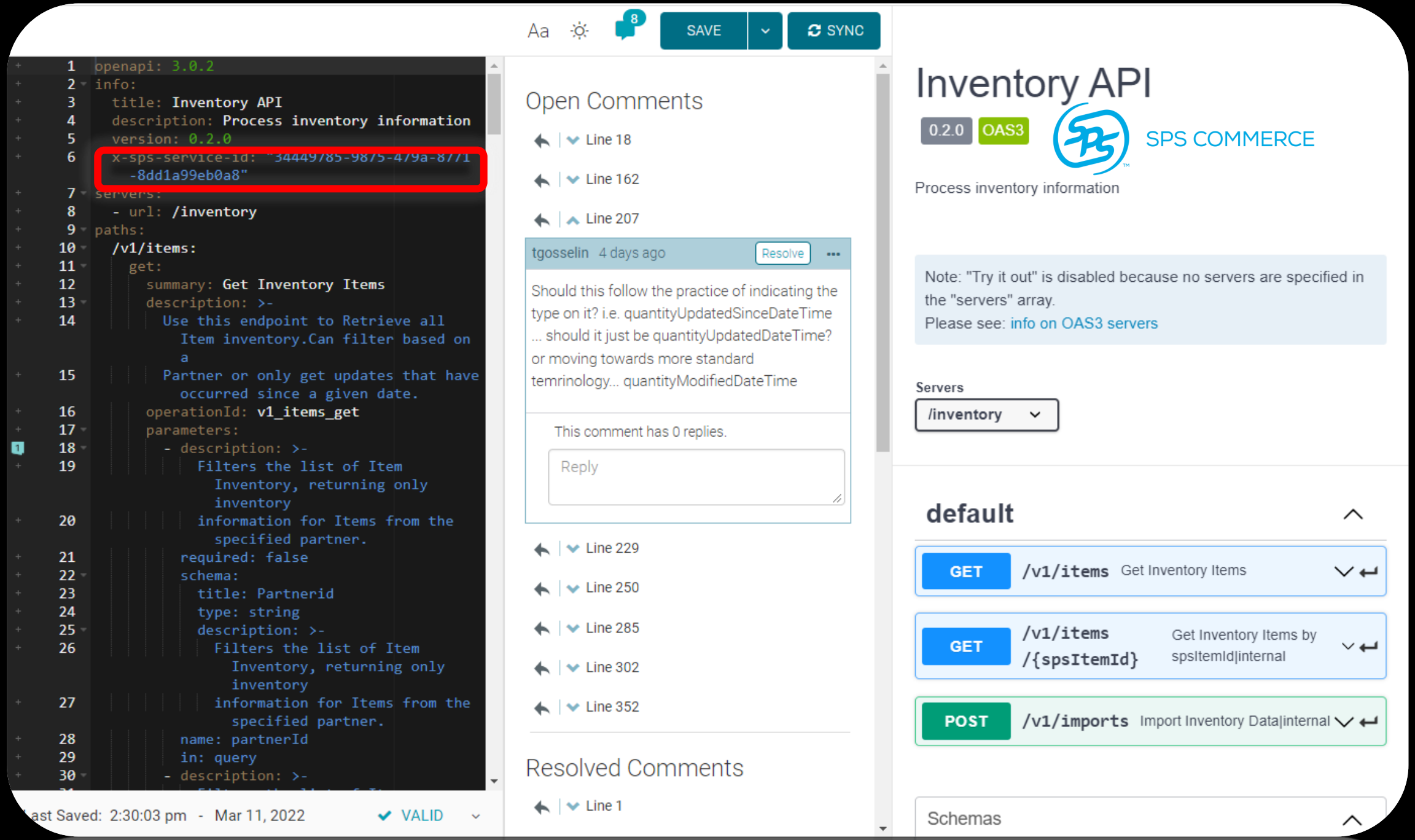
DESIGN

Automation








Aa
8
SAVE
SYNC
openapi: 3.0.2
info:
title: Inventory API
description: Process inventory information
version: 0.2.0
x-sps-service-id: "34449785-9875-479a-8771-8dd1a99eb0a8"
servers:
- url: /inventory
paths:
/v1/items:
get:
summary: Get Inventory Items
description: >-
Use this endpoint to Retrieve all Item inventory.Can filter based on a
Partner or only get updates that have occurred since a given date.
operationId: v1_items_get
parameters:
- description: >-
Filters the list of Item Inventory, returning only inventory
information for Items from the specified partner.
required: false
schema:
title: Partnerid
type: string
description: >-
Filters the list of Item Inventory, returning only inventory
information for Items from the specified partner.
name: partnerId
in: query
- description: >-
Last Saved: 2:30:03 pm - Mar 11, 2022
VALID
Open Comments
Line 18
Line 162
Line 207
tgosselin 4 days ago
Resolve
Should this follow the practice of indicating the type on it? i.e. quantityUpdatedSinceDateTime ... should it just be quantityUpdatedDateTime? or moving towards more standard temrinology... quantityModifiedDateTime
This comment has 0 replies.
Reply
Line 229
Line 250
Line 285
Line 302
Line 352
Resolved Comments
Line 1
Inventory API
0.2.0
OAS3
SPS COMMERCE
Process inventory information
Note: "Try it out" is disabled because no servers are specified in the "servers" array.
Please see: info on OAS3 servers
Servers
/inventory
default
GET /v1/items Get Inventory Items
GET /v1/items /{spsItemId} Get Inventory Items by spsItemId|internal
POST /v1/imports Import Inventory Data|internal
Schemas

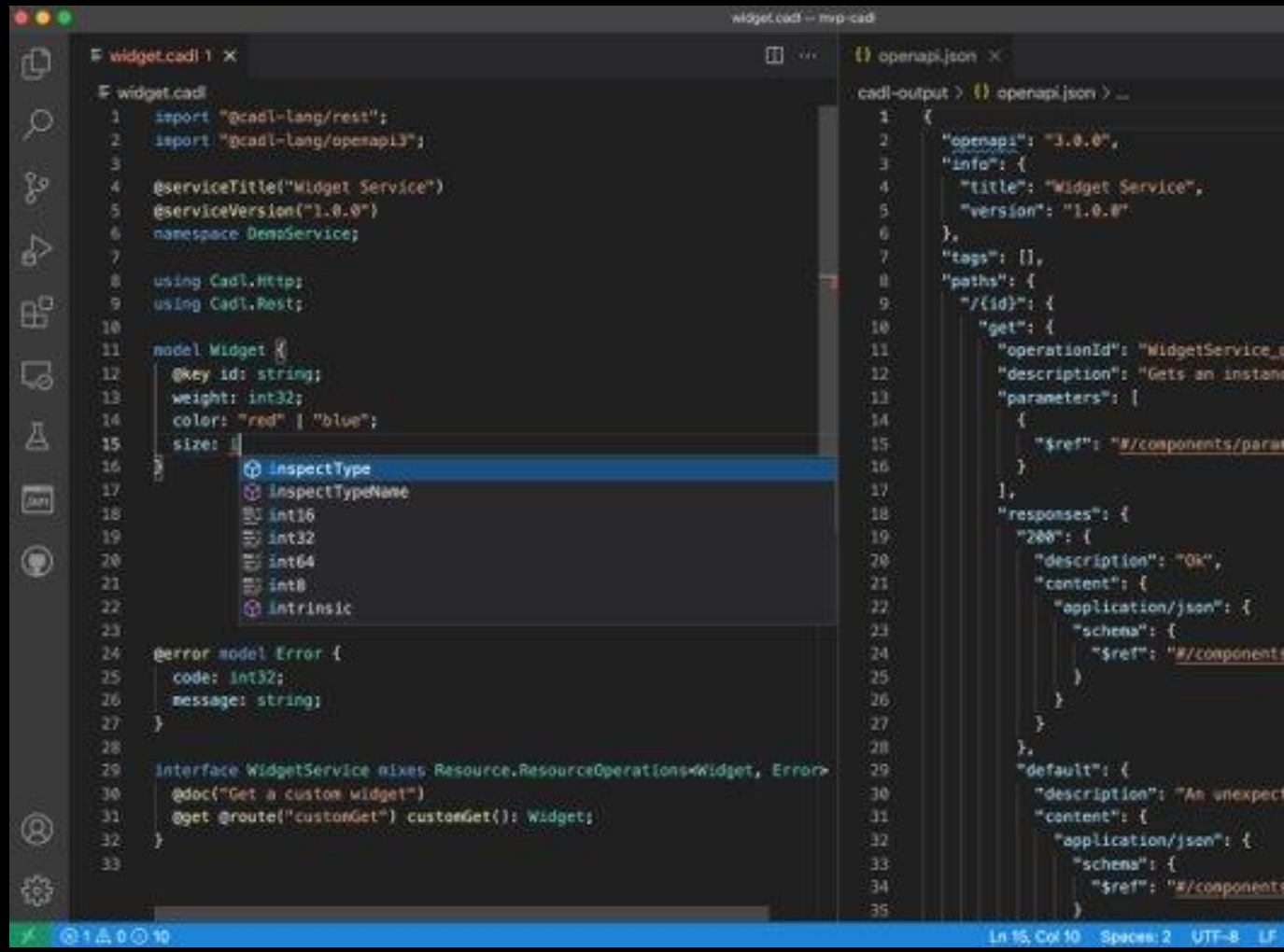
TypeSpec (Formerly: CADL)

https://microsoft.github.io/typespec/

Automation

https://github.com/microsoft/cadl

- A new language for designing APIs
- Describe API shapes common to REST, GraphQL, gRPC, and others.
- Highly extensible DSL
- Promote reuse through code sharing
- Version Aware
- Generate OpenAPI, Clients, Service Code, Documentation and other assets.
- Fully-integrated VSCode Extension



DEVELOPMENT

THE OLD STOMPING GROUND!

API Standards

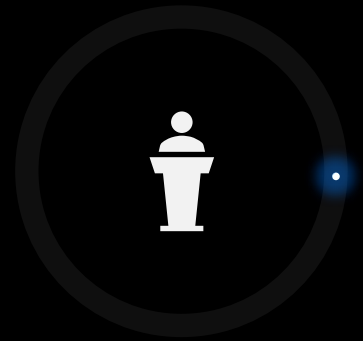
API Design

API Development

API Publishing

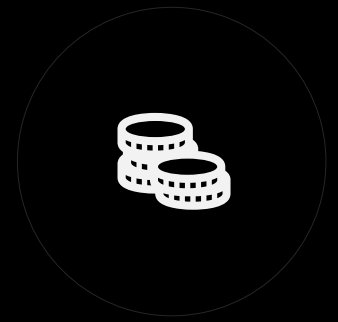
Contracts

Mocking, Stubs & SDKs



Interoperability

API Gateways, Reuse & Lifecycle



Validation

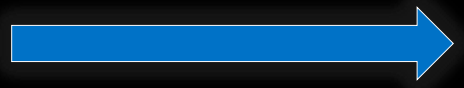
Did you build what you designed?

DEVELOPMENT

Contracts

Open API Document

```yaml
/v1/users:
  get:
    tags:
    - "Users"
    responses:
      "200":
        description: "Success"
        content:
          application/json:
            schema:
              type: "array"
              items:
                required:
                - "id"
                - "username"
                type: "object"
                properties:
                  id:
                    type: "integer"
                    format: "int32"
                  firstName:
                    type: "string"
                    nullable: true
                  lastName:
                    type: "string"
                    nullable: true
```



Mocking

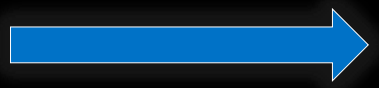
PRISM
Brought to you by Stoplight.

Static

Dynamic

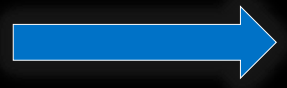
```
// REQUEST
GET https://api.mock.com/my-service/v1/users

// RESPONSE
[
    { “id”: 23, “firstName”: “John”, “lastName”: “Doe” }
]
```



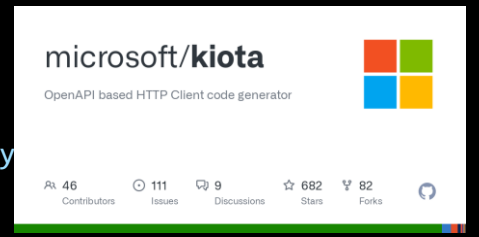
Stubs

```
[HttpGet]
[Route("/v1/users")]
[SwaggerOperation("V1UsersGet")]
[SwaggerResponse(statusCode: 200, type: typeof(List<InlineResponse200>), description: "Success")]
public virtual IActionResult V1UsersGet()
{
    var exampleJson = "[ {\n  \"firstName\" : \"firstName\",\n  \"lastName\" : \"lastName”…
    var example = exampleJson != null
          ? JsonConvert.DeserializeObject<List<InlineResponse200>>(exampleJson)
          : default(List<InlineResponse200>);    //TODO: Change the data returned
    return new ObjectResult(example);
}
```



SDKs

```
public interface IUsersApi : IApiAccessor
{
    List<InlineResponse200> V1UsersGet ();
    V1UsersBody2 V1UsersIdGet (int? id);
    void V1UsersIdPut (int? id, UsersIdBody
}
```



https://github.com/OpenAPITools/openapi-generator

DEVELOPMENT

Validation

```
================================================================================
==                            API CHANGE LOG                                  ==
================================================================================
                            Swagger Petstore
--------------------------------------------------------------------------------
--                             What's New                                     --
--------------------------------------------------------------------------------
- GET    /pet/{petId}

--------------------------------------------------------------------------------
--                           What's Deleted                                   --
--------------------------------------------------------------------------------
- POST   /pet/{petId}

--------------------------------------------------------------------------------
--                         What's Deprecated                                  --
--------------------------------------------------------------------------------
- GET    /user/logout

--------------------------------------------------------------------------------
--                           What's Changed                                   --
--------------------------------------------------------------------------------
- PUT    /pet
  Request:
        - Deleted application/xml
        - Changed application/json
          Schema: Backward compatible
- POST   /pet
  Parameter:
    - Add tags in query
  Request:
        - Changed application/xml
          Schema: Backward compatible
        - Changed application/json
          Schema: Backward compatible
```

HTML

Markdown

JSON

Design vs Actual (Generated and/or Runtime)

docker run openapitools/openapi-diff:latest **design.yml** **generated.json** **–fail-on-changed**

swagger

Build

V1.1.3

docker run openapi

#4.3.148 Security Scanning Demonstration (#197)

sps-ref-dotnetcore-api

Cancel

TEAM || Site Reliability Engineering (SRE) SLACK || travis-test-private SERVICE || d47ea086-64a1-4d88-af26-d222c3320c4e

Summary Tests Environments Artifactory Sauce Labs WhiteSource Bolt Build Report

Triggered by travisgosselin

View 20 changes

Repositories 2
SPSCommerce/sps-ref-dotnetcore-api , +1
See Sources card for details

Time started and elapsed
Tue at 9:09 PM
1d 14h 24m

Related
0 work items
2 published; 1 consumed

Tests and coverage
Get started

Warnings 1

API: Open API spec contains errors that DO NOT meet the SPS API Standards (https://docs.platform.spscommerce.com/api-design/standards/). See spectral log output above.
INTEGRATION • DOCS_DEPLOY INTEGRATION DEPLOY • publish documentation

1 approval needs review before this run can continue to PROD APPROVAL

View

Stages Jobs

CONTEXT
1 job completed 9s
1 artifact

COMPILE
3 jobs completed 3m 40s
100% tests passed
1 artifact

INTEGRATION
4 jobs completed 8m 40s
2 checks passed

PROD APPROVAL
Waiting
0/1 checks passed

PROD
Not started

DEVELOPMENT

Validation

Open API Document

```yaml
/v1/users:
  get:
    tags:
    - "Users"
    responses:
      "200":
        description: "Success"
        content:
          application/json:
            schema:
              type: "array"
              items:
                required:
                - "id"
                - "username"
                type: "object"
                properties:
                  id:
                    type: "integer"
                    format: "int32"
                  firstName:
                    type: "string"
                    nullable: true
                  lastName:
                    type: "string"
                    nullable: true
```

Build/Deploy

Runtime

REST API Fuzzer

GET /v1/users
DELETE /v1/users/xxxx

crunch

Microsoft *restler*

REST API Under Test

GET /v1/users -> 200 OK
DELETE /v1/users/xxxx -> 404 Not Found

DELETE /v1/users/xxxx

Request

REST API Validation Proxy

REST API

PRISM
Brought to you by Stoplight.

crunch

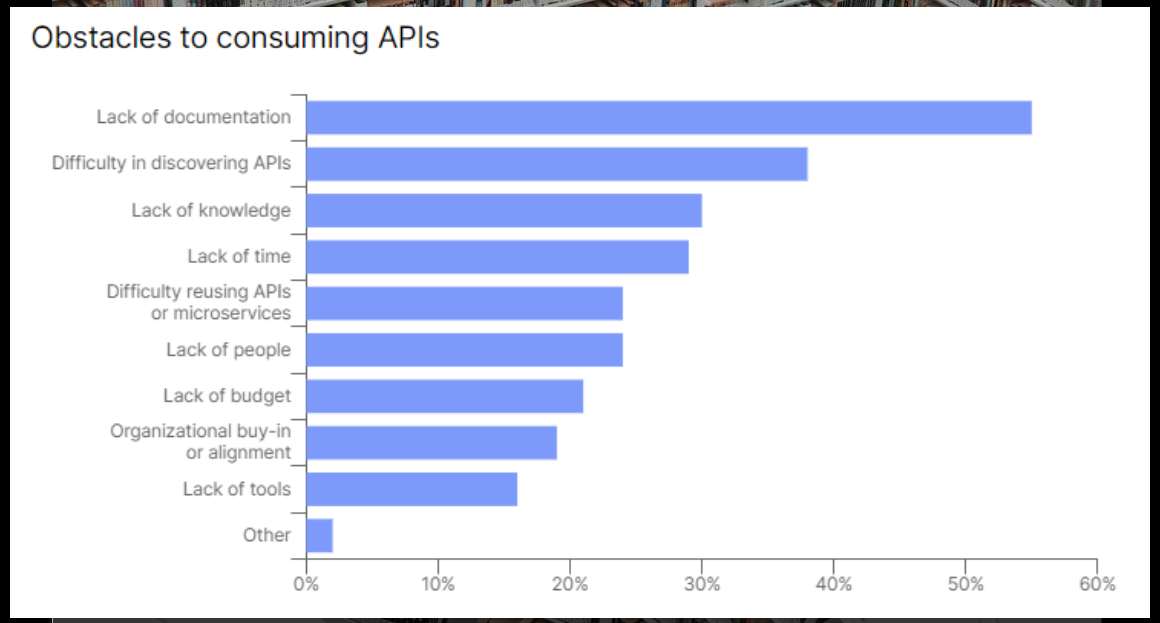
openapi-cop

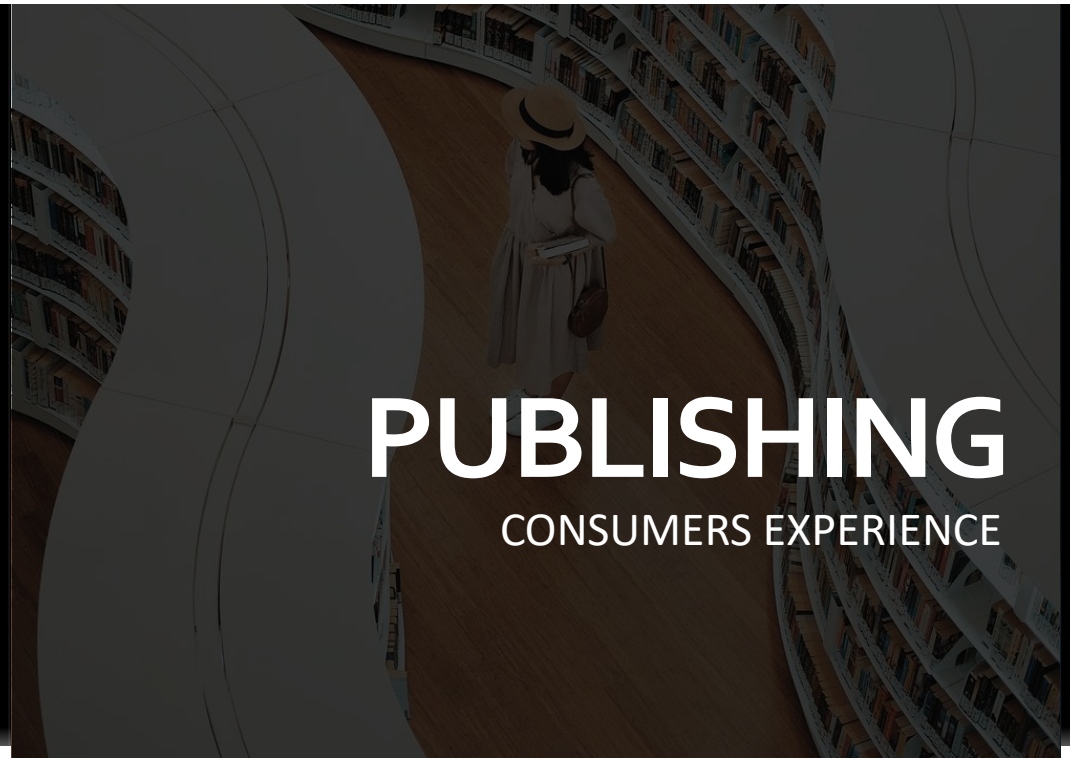
DELETE /v1/users/xxxx -> 500

Logging / Report

Enforcement

postman.com/state-of-api





PUBLISHING

CONSUMERS EXPERIENCE

- API Standards
- API Design
- API Development
- API Publishing

API Catalog

Documentation

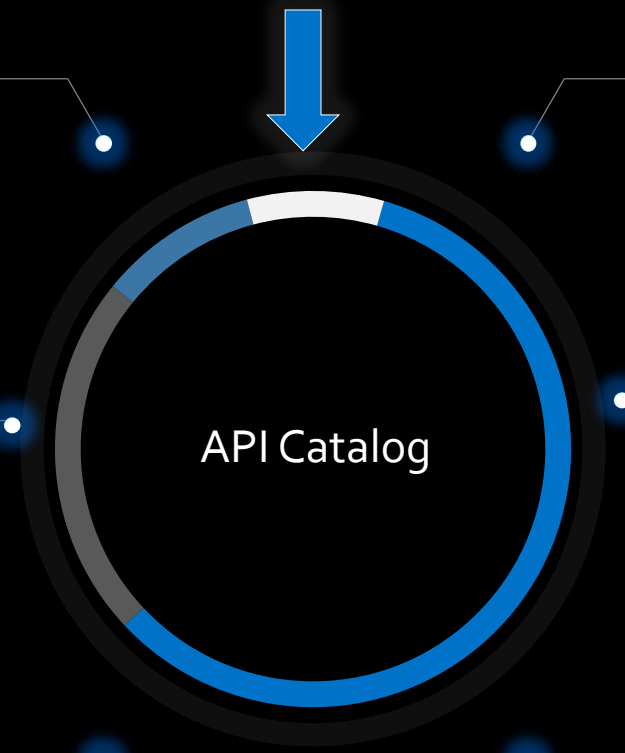
Organization

> "Great API documentation: That's a discipline that's really lacking..."
>
> James H.

PUBLISHING

API Catalog

Open API Document



Centralized

Eliminate Distributed API Specs, Inventory of all APIs

Documentation

Guides, Tutorials & Examples

Discoverable

Searchable, APIs & Endpoints, Critical for Internal Communication

Versioned

Change Strategy & Immutable Versions Over Time

Developer Portal

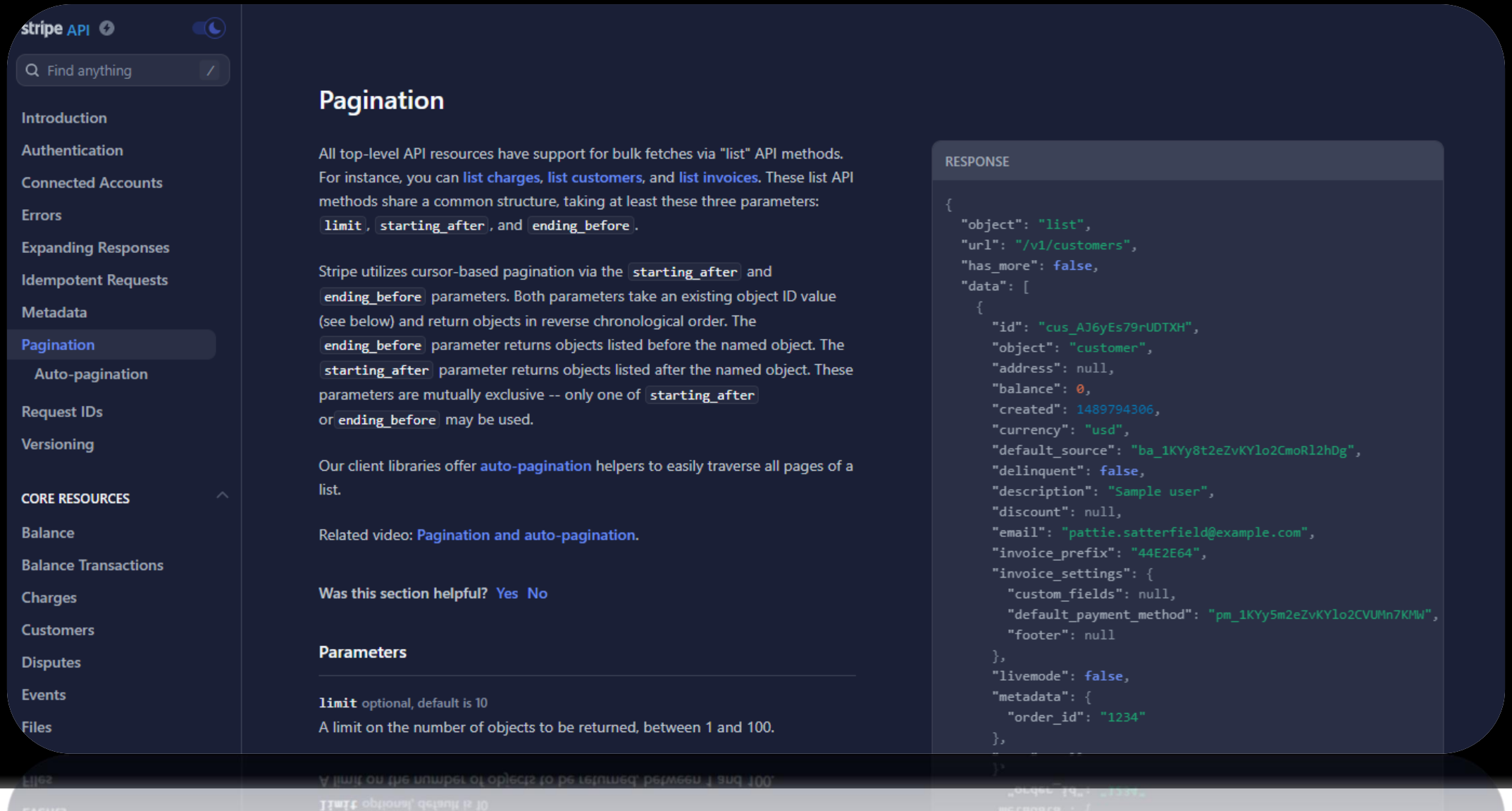
Internal, External, Partners, Interactive, Communication with Developers

API

Programmatically Accessible through Automation

PUBLISHING

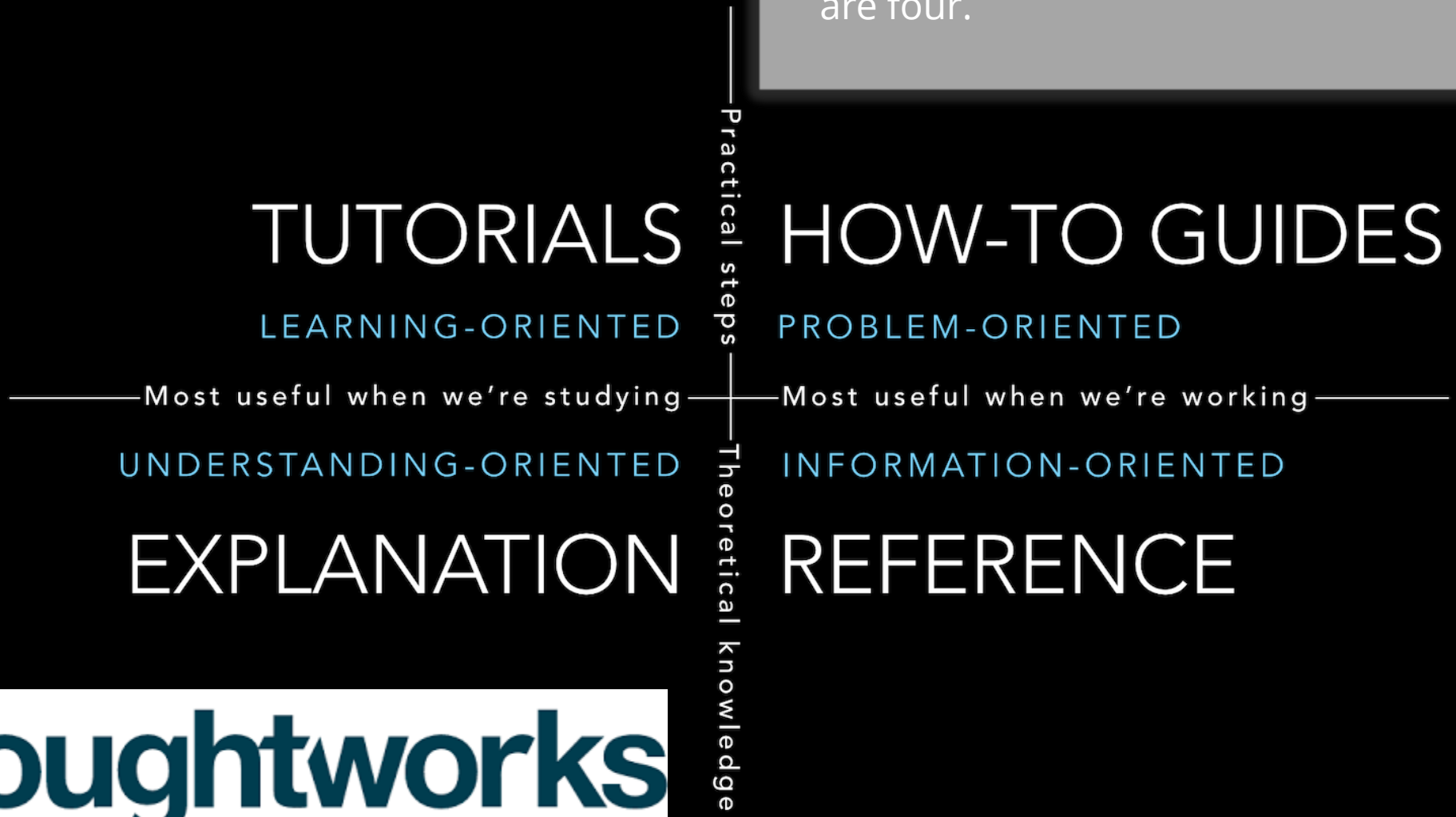
Great Documentation Examples



PUBLISHING

Documentation Types

> "There is a secret that needs to be understood in order to write good software documentation: there isn't one thing called documentation, there are four."
>
> Divio

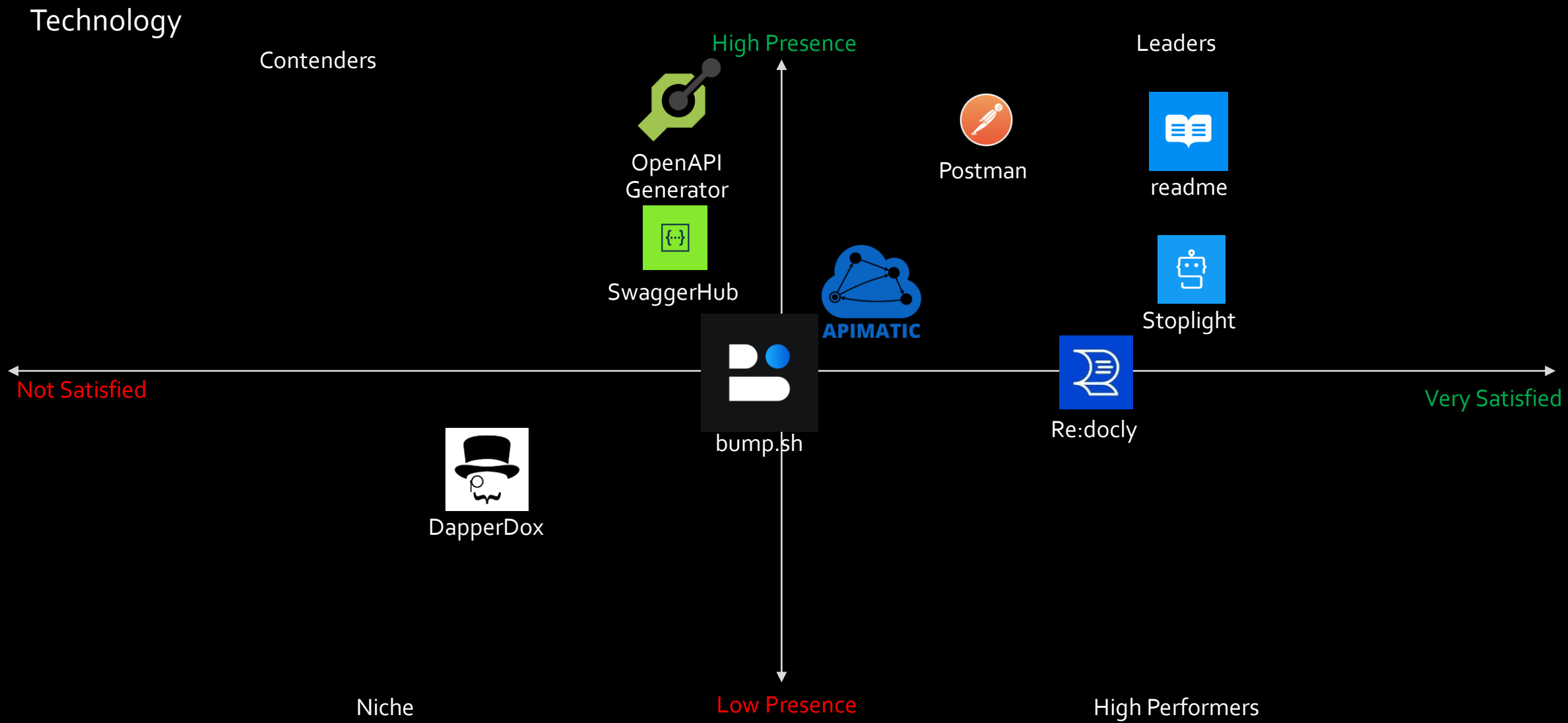


/thoughtworks

TECH RADAR (2022): DOCUMENTATION QUADRANTS

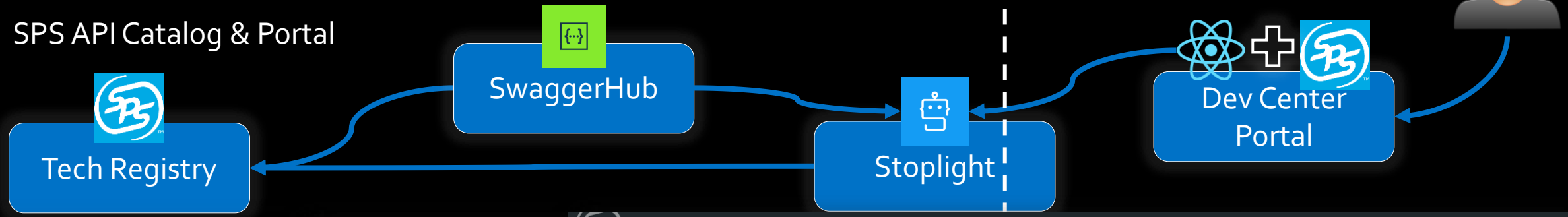
https://documentation.divio.com/

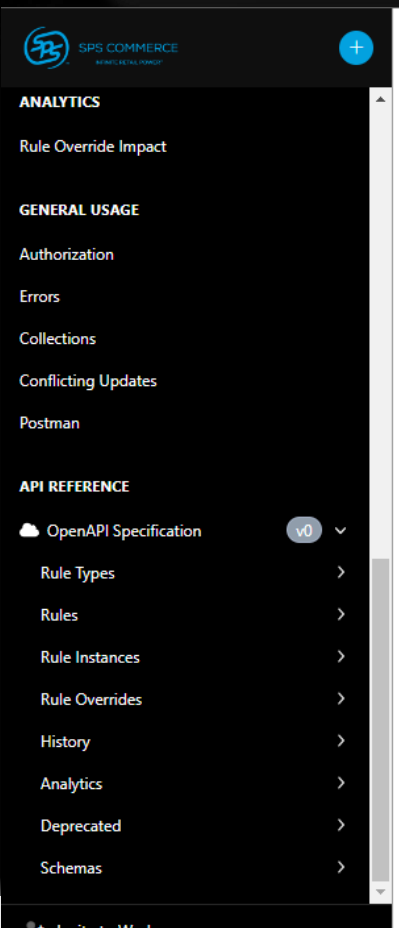
PUBLISHING

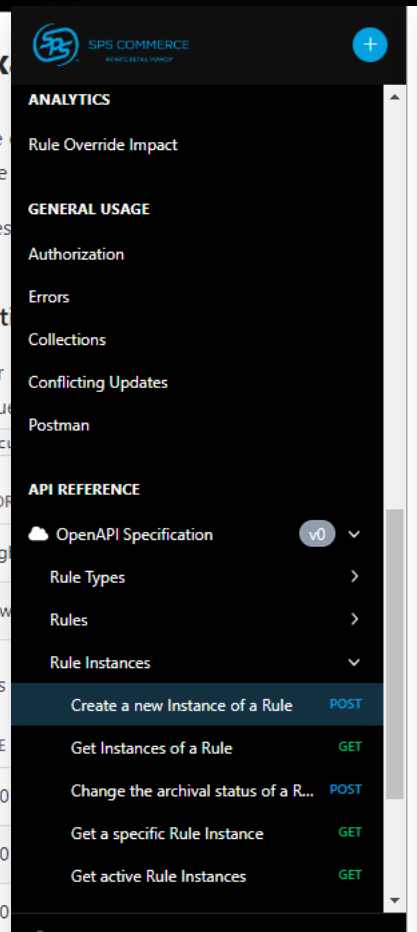

Technology
Contenders
High Presence
Leaders
OpenAPI Generator
Postman
readme
SwaggerHub
APIMATIC
Stoplight
Not Satisfied
bump.sh
Re:docly
Very Satisfied
DapperDox
Niche
Low Presence
High Performers

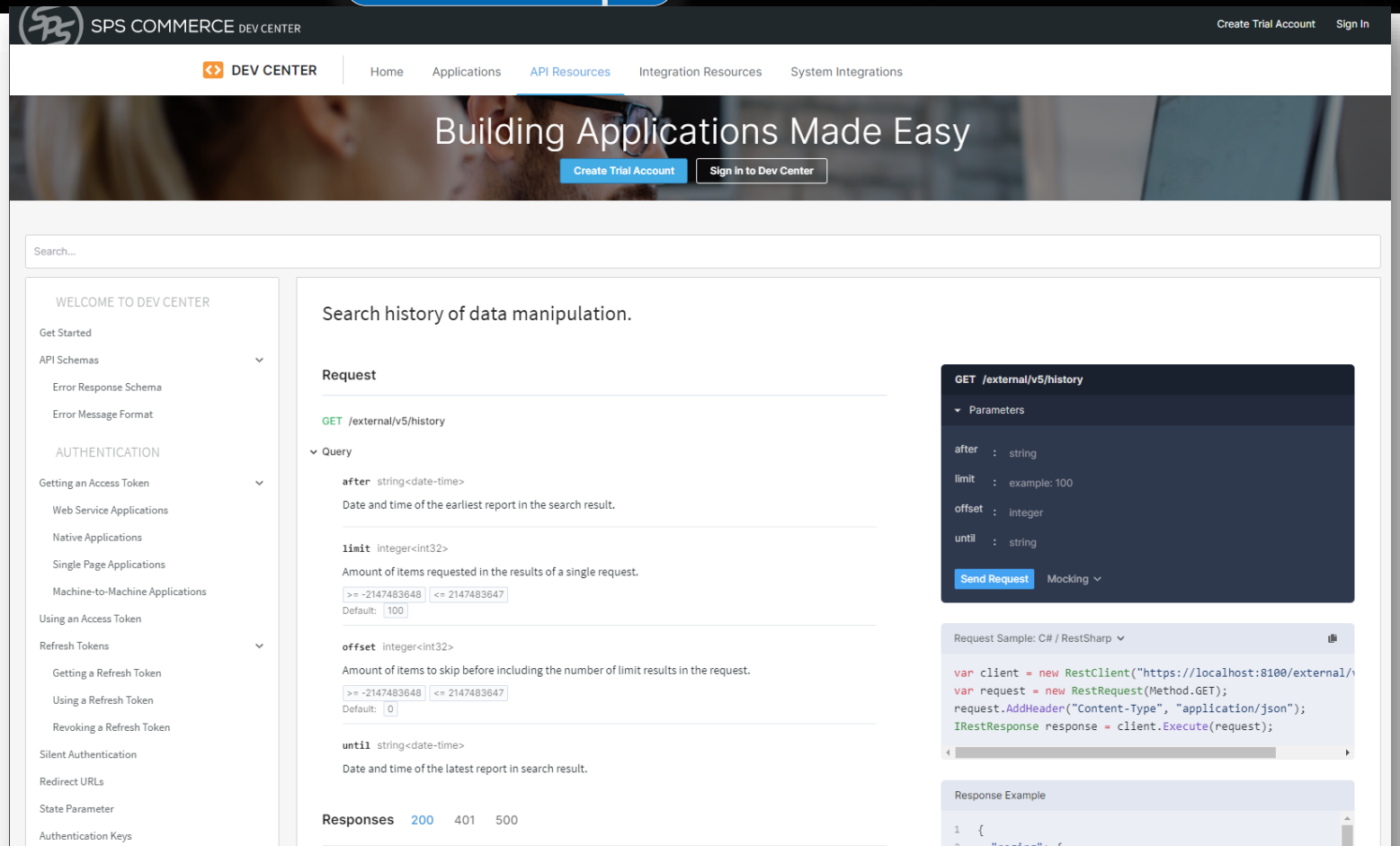
PUBLISHING

SPS API Catalog & Portal





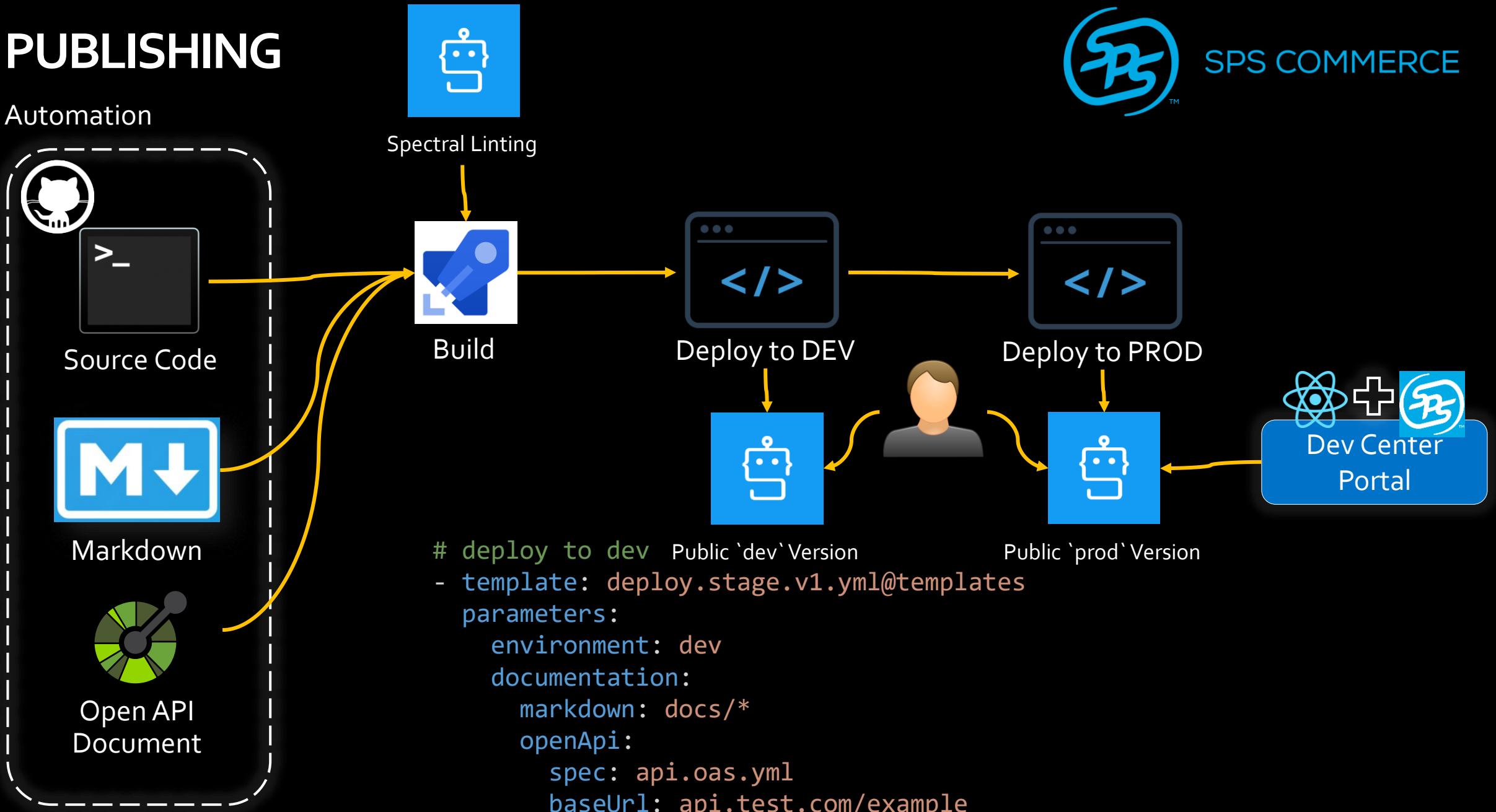




PUBLISHING

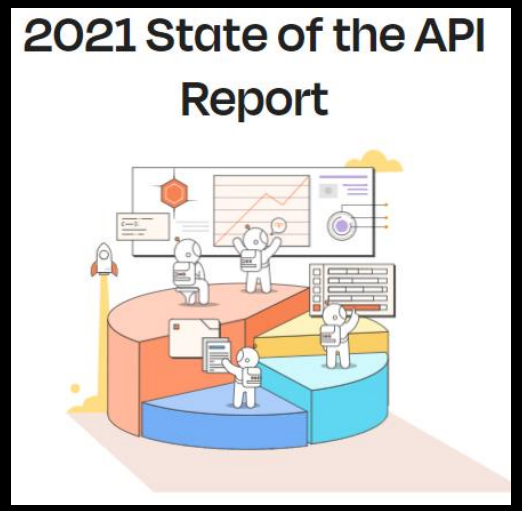
SPS COMMERCE

Automation




```
# deploy to dev
- template: deploy.stage.v1.yml@templates
  parameters:
    environment: dev
    documentation:
      markdown: docs/*
      openApi:
        spec: api.oas.yml
        baseUrl: api.test.com/example
```

RESOURCES



postman.com/state-of-api/

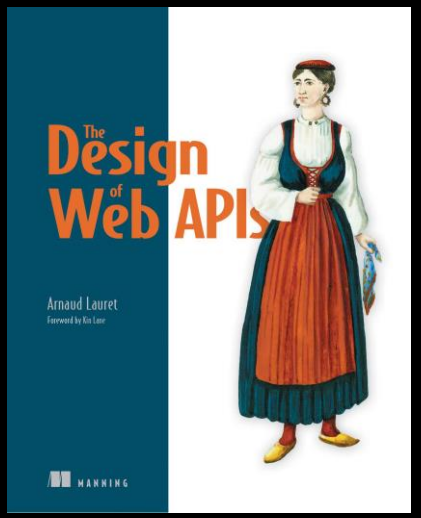
https://stoplight.io/

SPECTRAL
Brought to you by Stoplight.

Stoplight

API ROADMAP

APIsecurity.io



42 crunch

smartbear.com/resources/ebooks/the-state-of-api-2020-report/

OpenAPI.Tools

We want to keep API developers up to date with the best OpenAPI tooling around, and help direct folks to high quality modern tooling, instead of being stuck on old v2-based rubbish.

Contribute | APIs You Won't Hate

SPS COMMERCE

github.com/SPSCommerce/sps-api-standards

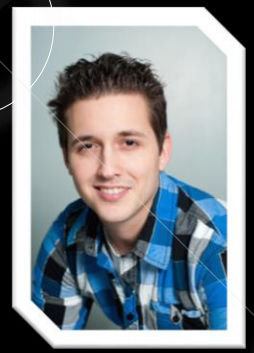
> **APIs Are Forever**
>
> Once an API is created, it should never be deleted, or changed. "Once you put an API out there, businesses will build on top of it," Vogels said, adding that changing the API will basically break their businesses.
>
> Werner Vogel's – 6 Best Practices for API Design

IS YOUR API READY FOR FOREVER?

- Alignment
- Enablement
- Collaboration
- Guidance

ARE YOU SUCCEEDING IN API DESIGN AT SCALE?

API DESIGN FIRST SUCCEEDING WITH API GOVERNANCE



TRAVIS GOSSELIN

travisgosselin.com

linkedin.com/in/travisgosselin

@travisjgosselin