

APIs:

CODE FIRST

DESIGN FIRST

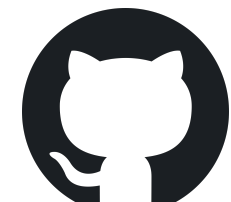
vs



LET'S FIGHT!



Sébastien Charrier



@scharrier

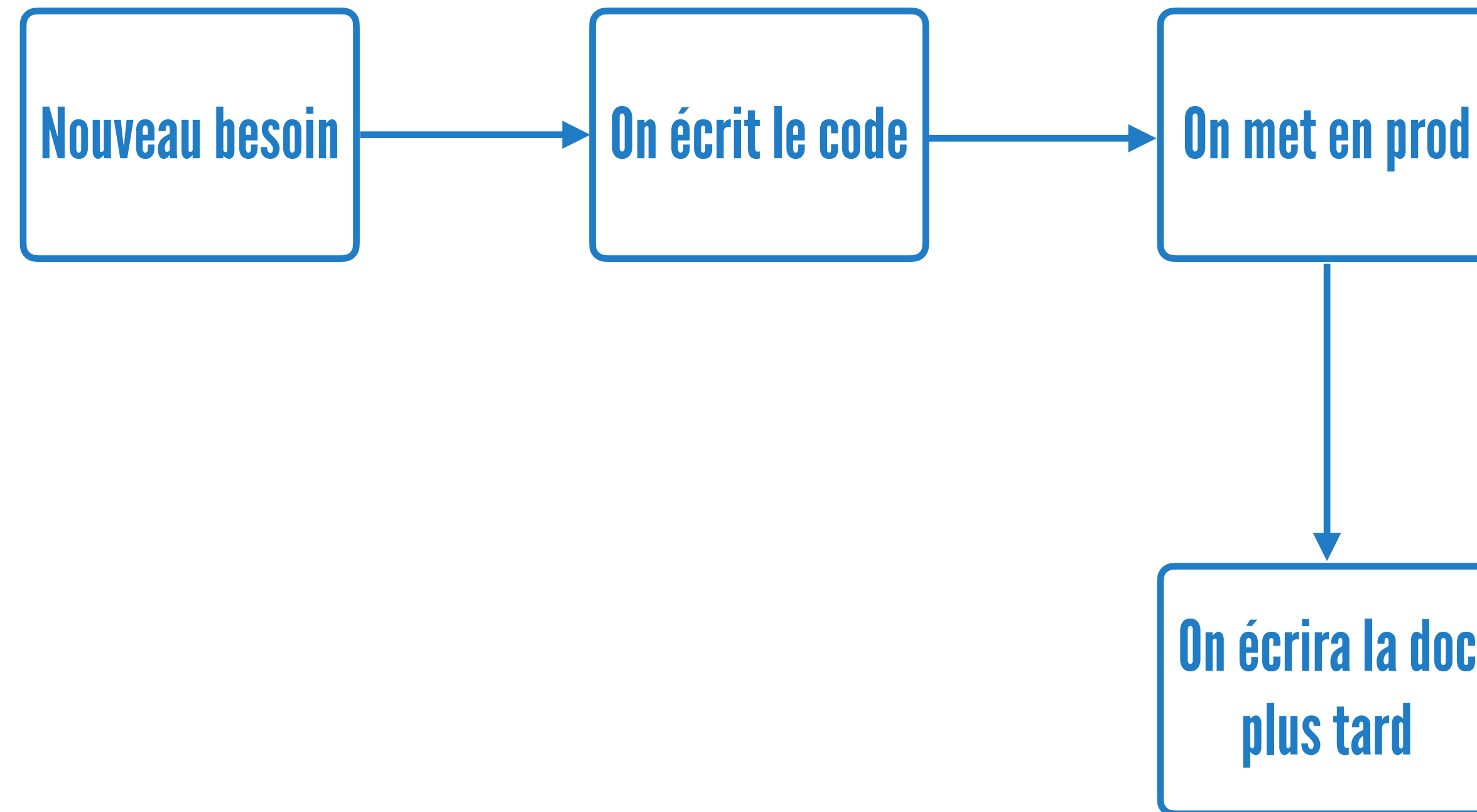


<https://bump.sh>



**QUI UTILISE DES APIS NON
DOCUMENTÉES ? 🙅**

**L'APPROCHE
CODE AND
DOCUMENT LATER**



””

**EST-CE QUE TU PEUX
ÉCRIRE LA DOC EN
URGENCE ?**





MINISTÈRE DE L'ÉDUCATION NATIONALE

3.3.2. REPONSE



API_Association_xml
_Schema.xml

Sur la base d'une requête unitaire, l'API interroge les différents fournisseurs d'informations et transmet l'information complétée au demandeur :

- Données validées du RNA et/ou du répertoire SIRENE.
- Données administratives complémentaires issues de DBAsso (origine : flux DAC) ;
- L'index des documents.

Après l'authentification, les contrôles sont effectués dans l'ordre chronologique suivant :

Id	Code	Réponse
Id conforme	200	La réponse contient un tableau détaillant les caractéristiques de l'association. L'objet en réponse respecte la structure suivante. <pre>{ "nom": xxxxxxxxxxxx, "sigle": xxxxxxxxxxxx, "id_rna": xxxxxxxxxxxx, "id_siren": xxxxxxxxxxxx, "id_siret_siege": xxxxxxxxxxxx, "date_pub_jo": xx/xx/xxxx [...] }</pre>
Id format non conforme		<code>{"error": "id n'est pas reconnu comme un code acceptable"}</code>
Id non existant		<code>{"error": "id not found in dbasso"}</code>
Méthode non conforme		<code>{"error": " Method Not Allowed"}</code>
	500	Internal Server Error

PLUS TARD

=

JAMAIS

A woman wearing a grey headscarf and a long-sleeved grey dress is walking through a crowd of people in a stone-walled courtyard. She has a serious expression. The background shows other people, some men in brown and dark clothing, and a stone wall with an arched doorway. The scene is lit with natural light, creating soft shadows.

SHAME. SHAME. SHAME.

**AUJOURD'HUI NOUS
AVONS LES OUTILS**

API SPECIFICATIONS

FTW 🙌

« AN API SPECIFICATION PROVIDES A BROAD UNDERSTANDING OF HOW AN API BEHAVES AND HOW THE API LINKS WITH OTHER APIS. IT EXPLAINS HOW THE API FUNCTIONS AND THE RESULTS TO EXPECT WHEN USING THE API »»

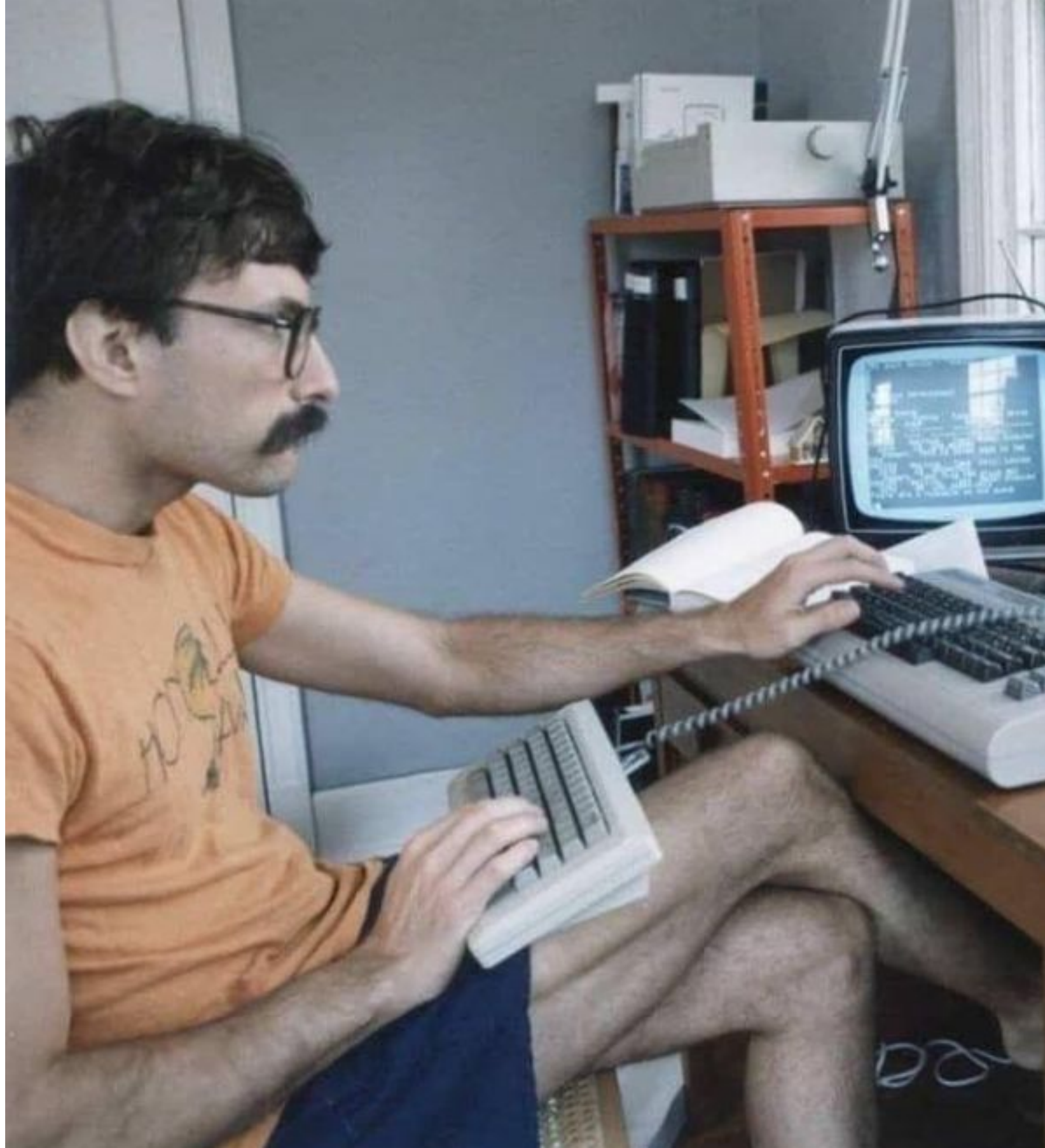
<https://swagger.io/resources/articles/difference-between-api-documentation-specification/>

Contract

« AN API ~~SPECIFICATION~~ PROVIDES A BROAD UNDERSTANDING OF HOW AN API BEHAVES AND HOW THE API LINKS WITH OTHER APIS. IT EXPLAINS HOW THE API FUNCTIONS AND THE RESULTS TO EXPECT WHEN USING THE API »»

<https://swagger.io/resources/articles/difference-between-api-documentation-specification/>

SOAP



APIS REST

**OPENAPI, RAML,
BLUEPRINT**

APIS EVENT-DRIVEN

ASYNCAPI 

<https://asyncaapi.com>

GRAPHQL

PROTOBUF

**ON PEUT FAIRE TOUT UN
TAS DE TRUCS AVEC**

**ET AVEC LES BONS OUTILS
ON PEUT LE FAIRE
AUTOMATIQUEMENT**



DOCUMENTATION

The screenshot shows a web browser window with the URL `bump.sh`. The page title is "Create a new version" and it includes a "Basic token" badge. The main content area describes the endpoint and lists parameters: `documentation` (STRING), `hub` (STRING), `documentation_name` (STRING), `auto_create_documentation` (BOOLEAN), and `definition` (REQUIRED / STRING). A code block on the right shows a curl command and a JSON request example. The left sidebar contains a navigation menu with sections for TOPICS, ENDPOINTS, and Versions.

Bump Changelog Download source

TOPICS

- Introduction
- Authentication

ENDPOINTS

- Ping >
- Previews >
- Validations >
- Versions >

POST [Create a new version](#)

GET [Fetch a full documentation version including diff summary](#)

Create a new version

Basic token

Deploy a new version for a given documentation, which will become the current version.

BODY

documentation STRING

UUID or slug of the documentation.

hub STRING

UUID or slug of the hub if the documentation is part of a hub.

documentation_name STRING

Name of the documentation to create. Used only if `auto_create_documentation` is set.

auto_create_documentation BOOLEAN

Create the documentation if it does not exist yet. Must be used with a `hub` and a `documentation`.

definition REQUIRED / STRING

Serialized definition of the version. This should be an OpenAPI 2.x, 3.x or

POST /versions

```
$ curl \
-X POST https://bump.sh/api/v1/versions \
-H "Content-Type: application/json" \
-d '{"documentation": "0776d85d-e097-47c1-8c60-cb1190d11945", "hub": "my_hu
```

Request example

```
{
  "documentation": "0776d85d-e097-47c1-8c60-cb1190d11945",
  "hub": "my_hub_slug",
  "documentation_name": "string",
  "auto_create_documentation": true,
  "definition": "{ \"openapi\": \"3.0\", \"info\": { \"title\": ... }}\n",
  "references": [
    {
      "location": "https://example.com/api/models/pet.yml",
      "content": "string"
    }
  ],
  "previous_version_id": "3ef8f52f-9056-4113-840e-2f7183b90e06",
  "unpublished": false
}
```


VALIDER LE COMPORTEMENT DE L'API



```
it 'successfully gets a device' do
  device = create(:device)

  get("/devices/#{device.id}")

  expect(response).to have_http_status(:ok)
  expect(response).to match_response_schema('device')
end
```

MOCKING



GÉNÉRATION DES CLIENTS ET MODÈLES

The screenshot shows a web browser window with the URL `openapi-generator.tech`. The page title is "Generators List". The navigation menu includes "Getting Started", "Roadmap", "FAQ", "Team", "Blog", and "API". The left sidebar lists "Getting Started", "Extending", "Contributing", "About", "Releases", "API", and "Generators List". The main content area features the heading "Generators List" and the text "The following generators are available:". Below this, there is a section for "CLIENT generators" with a list of 15 items: `ada`, `android`, `apex`, `bash`, `c`, `clojure`, `cpp-qt-client`, `cpp-restsdk`, `cpp-tiny (beta)`, `cpp-tizen`, `cpp-ue4 (beta)`, `crystal (beta)`, `csharp`, and `csharp-dotnet2 (deprecated)`. On the right side, there is a search bar and a list of generator categories: "CLIENT generators", "SERVER generators", "DOCUMENTATION generators", "SCHEMA generators", and "CONFIG generators".

OpenAPI Generator

Getting Started

Generators Roadmap FAQ Team Blog API

Search

Generators List

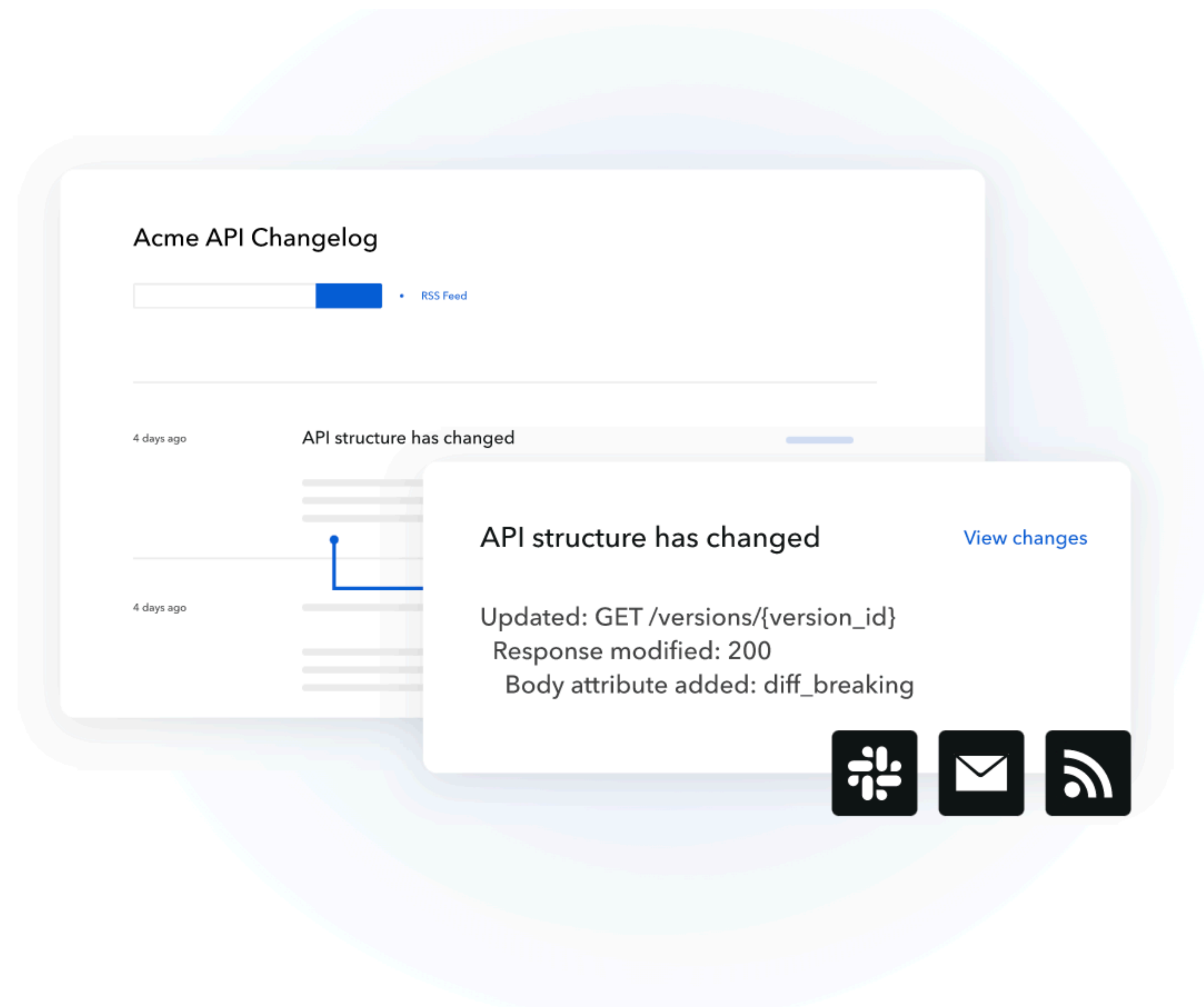
The following generators are available:

CLIENT generators

- `ada`
- `android`
- `apex`
- `bash`
- `c`
- `clojure`
- `cpp-qt-client`
- `cpp-restsdk`
- `cpp-tiny (beta)`
- `cpp-tizen`
- `cpp-ue4 (beta)`
- `crystal (beta)`
- `csharp`
- `csharp-dotnet2 (deprecated)`

CLIENT generators
SERVER generators
DOCUMENTATION generators
SCHEMA generators
CONFIG generators

SYNCHRONISATION DES CHANGEMENTS



COMMENT ON LE CRÉE ?

CODE FIRST



DESIGN FIRST

L'APPROCHE CODE FIRST





```
class ReportController {  
    @OpenApi(  
        path = "/reports",  
        method = HttpMethod.POST,  
        // ...  
    )  
    public static void createReport(Context ctx) {  
        // ...  
    }  
}
```

L'APPROCHE

CODE FIRST

THEN ANNOTATE*



DOCUMENTATION



VALIDATION DES DONNEES



MOCKING



GÉNÉRATION DES CLIENTS



SYNCHRONISATION DES CHANGEMENTS



 **DOCUMENTATION**

 **VALIDATION DES DONNEES**

 **MOCKING**

 **GÉNÉRATION DES CLIENTS**

 **SYNCHRONISATION DES CHANGEMENTS**

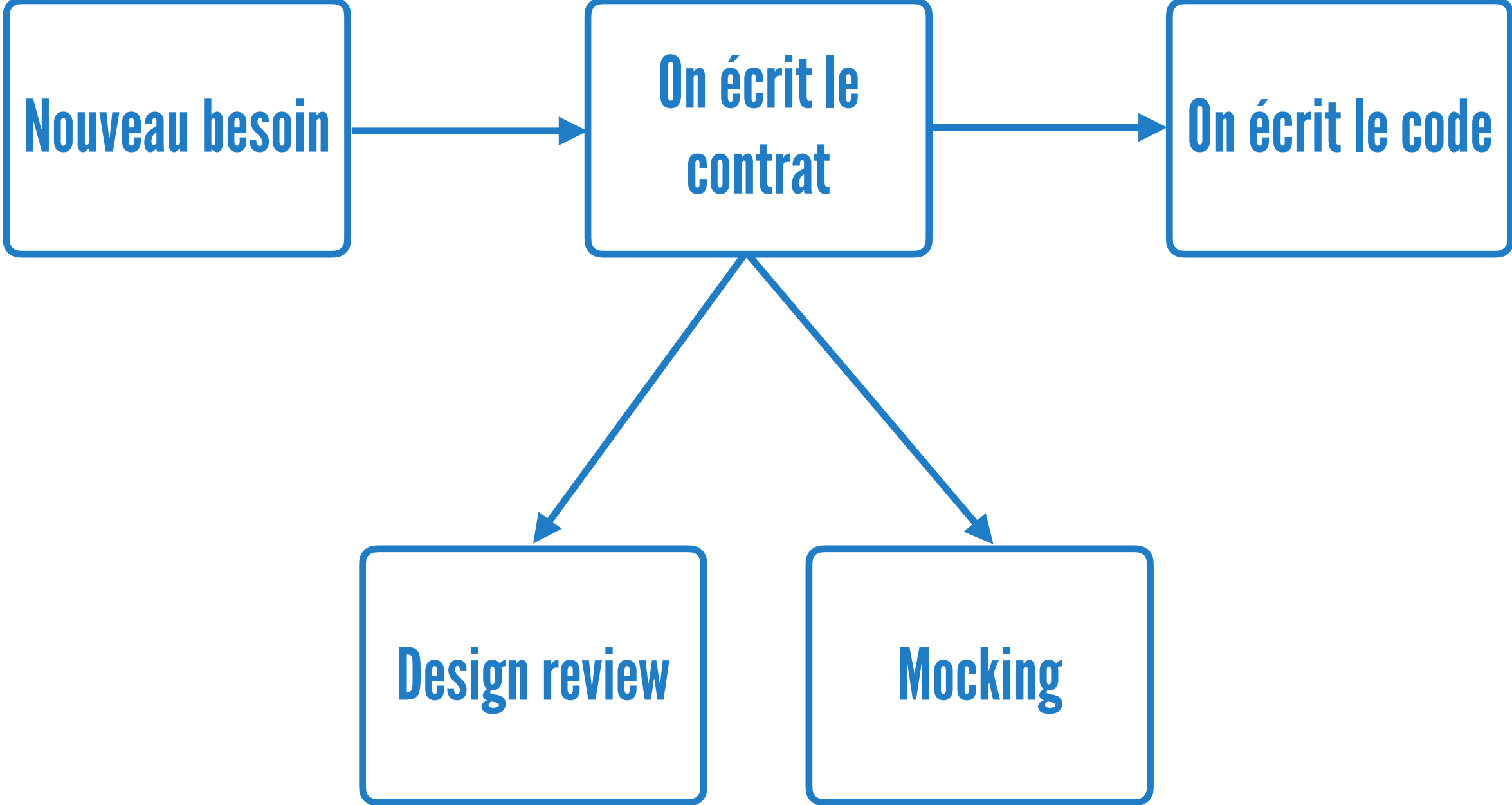
**VOUS DEVEZ
SYNCHRONISER TOUT
UN ÉCOSYSTÈME
AUTOUR DE VOTRE API.**



**L'APPROCHE
DESIGN FIRST**

PRINCIPE :

**LE CONTRAT EST LA
SOURCE UNIQUE DE
VÉRITÉ**



STOPLIGHT STUDIO

The screenshot displays the Stoplight Studio interface for an API endpoint. The top navigation bar includes a project name 'studio-demo', a '+ Commit' button, and tabs for 'Todos' and 'Operation'. The left sidebar shows a tree view for 'Swagger Petstore' with folders for 'Paths', 'Models', and 'Embedded Models'. Under 'To-dos', the selected path is '/todos/{todoid}' with methods GET, PUT, and DELETE. Below it, another path '/todos' has methods GET and POST. The main area is titled 'Get Todo' and shows the URL 'http://todos.stoplight.io /todos/{todoid}'. A row of method buttons is visible, with 'PUT' highlighted. The endpoint description is currently blank. Below the description are buttons for '+ Security', '+ Header', and '+ Query Param'. There are also buttons for 'JSON Request Body' and 'Form Data'. A 'Response' section shows a list of status codes: 200 (selected), 404, and 500. The response description is also blank. At the bottom, there are tags 'my-sug' and 'remove tags', and a status indicator '7 Issues'.

studio-demo + Commit Todos Operation

APIs Docs Files

Swagger Petstore

- Paths
- Models
- Embedded Models

To-dos

- Paths
 - /todos/{todoid} GET PUT DELETE
 - /todos GET POST
- Models

Common

- Models

Get Todo

http://todos.stoplight.io /todos/{todoid}

GET POST PUT PATCH DELETE HEAD OPTIONS TRACE

Endpoint description...

+ Security + Header + Query Param

JSON Request Body Form Data

+ Response 200 404 500 200

Response description...

Headers +

my-sug remove tags 7 Issues

INSOMNIA

The screenshot displays the Insomnia API client interface for the Fathom API 1.0. The interface is divided into several sections:

- Dashboard / Fathom API 1.0**: The top navigation bar includes tabs for **DESIGN**, **DEBUG**, and **TEST**. It also features a **Preview: On** indicator, a **Setup Git Sync** button, and user profile icons.
- INFO**: A sidebar menu with a search icon.
- SERVERS**: A list of servers, currently showing `https://api.usefathom.com/v1/`.
- PATHS**: A list of API endpoints with associated HTTP methods:
 - `/account` (GET)
 - `/sites` (GET, POST)
 - `/sites/{site_id}` (GET, POST, DELETE)
 - `/sites/{site_id}/data` (DELETE)
- PARAMETERS**: A section for defining request parameters.
- SCHEMAS**: A section for defining request and response schemas.

The main workspace displays the OpenAPI specification for the `POST /sites/{site_id}` endpoint. The code is as follows:

```
112           $ref: "#/components/schemas/BaseSite"
113     post:
114       summary: Create Site
115       description: Create a site.
116       requestBody:
117         content:
118           "application/x-www-form-urlencoded":
119             schema:
120               allOf:
121                 - $ref: "#/components/schemas/BaseSite"
122                 - properties:
123                   share_password:
124                     type: string
125                     description: When sharing is set to private, you must also
126       responses:
127         "201":
128           description: A Site object.
129           content:
130             application/json:
131               schema:
132                 $ref: "#/components/schemas/RespondedSite"
133     /sites/{site_id}:
134     get:
135       summary: Get Site
136       description: Return a single site.
137       parameters:
138         - $ref: "#/components/parameters/SiteId"
139       responses:
140         "200":
141           description: A Site object.
142           content:
143             application/json:
144               schema:
145                 $ref: "#/components/schemas/RespondedSite"
146     post:
147       summary: Update Site
148       description: Update a site.
149       parameters:
150         - $ref: "#/components/parameters/SiteId"
151       requestBody:
152         content:
153           "application/x-www-form-urlencoded":
154             schema:
155               allOf:
156                 - $ref: "#/components/schemas/BaseSite"
157                 - properties:
158                   share_password:
159                     type: string
160                     description: When sharing is set to private, you must also
161       responses:
162         "200":
163           description: A Site object.
```

The right-hand panel provides a summary of the API:

- Fathom API 1.0 OAS3**: The API title and version.
- Description**: "The Fathom Analytics API allows you to take complete control of your Fathom account. You can control all entities within the account, generate ad-hoc custom reports and even use server-side tracking. It's an incredibly powerful API, and we've tried to cater to every possible use case."
- Usage Note**: "Each API request counts against your total monthly pageviews. So if you make 10 Create Site requests, that will count in the same way that 10 pageviews would. We do this because we want to give you the flexibility to use our API how you wish, and we want to ensure it's sustainable without us having to raise plan prices."
- Disclaimer**: "When running complex aggregation queries that are unique (haven't been run before by any user), you will experience a slight lag for the first time you run the query. This is because our database engine has to create a 'query plan', to find the most optimal way to run what you're asking for. After that, it will be rapid."
- Contact**: "If there's anything you think we should add, please let us know at support@usefathom.com."
- Servers**: A dropdown menu showing the current server: `https://api.usefathom.com/v1/`.
- default**: A button to view the default configuration.
- Schemas**: A dropdown menu showing the selected schema: `RespondedSite`.

ASYNCAPI STUDIO

The screenshot displays the AsyncAPI Studio interface. The browser address bar shows `studio.asyncapi.com`. The top navigation bar includes the AsyncAPI logo and the text "AsyncAPI studio beta".

Left Sidebar:

- Information
- Servers
 - KAFKA-SECURE test
- Operations
 - PUB smartylighting.streetlights...
 - SUB smartylighting.streetlights...
 - SUB smartylighting.streetlights...
 - SUB smartylighting.streetlights...
- Messages
 - lightMeasured
 - turnOnOff
 - dimLight
- Schemas
 - lightMeasuredPayload
 - turnOnOffPayload
 - dimLightPayload
 - sentAt

Main Editor:

The editor shows a YAML document titled "From localStorage". The content is as follows:

```
1 asyncapi: '2.2.0'
2 info:
3   title: Streetlights Kafka API
4   version: '1.0.0'
5   description: |
6     The Smartylighting Streetlights API allows you to
7     remotely manage the city lights.
8
9     ### Check out its awesome features:
10
11     * Turn a specific streetlight on/off 🏠
12     * Dim a specific streetlight 🌙
13     * Receive real-time information about
14     environmental lighting conditions 📊
15
16   license:
17     name: Apache 2.0
18     url: https://www.apache.org/licenses/LICENSE-2.0
19
20   servers:
21     test:
22       url: test.mykafkacluster.org:8092
23       protocol: kafka-secure
24       description: Test broker
25       security:
26         - saslScram: []
27
28   defaultContentType: application/json
29
30   channels:
31     smartylighting.streetlights.1.0.event.{streetlightId}
32     .lighting.measured:
33       description: The topic on which measured values
```

Event Visualiser:

The Event Visualiser is titled "Streetlights Kafka API". It shows the application details and a list of servers. The "OUT" section displays three subscribed events:

- Event 1:** `smartylighting.streetlights.1.0.action.{streetlightId}.turn.on`. Messages include a `turnOnOff` button.
- Event 2:** `smartylighting.streetlights.1.0.action.{streetlightId}.turn.off`. Messages include a `turnOnOff` button.
- Event 3:** `smartylighting.streetlights.1.0.action.{streetlightId}.dim`. Messages include a `dimLight` button.

Bottom Panel:

The bottom panel shows "PROBLEMS 0" and "VALID YAML". A message states: "No problems have been detected in the AsyncAPI document so far."

- ✓ **DOCUMENTATION**
- ✓ **VALIDATION DES DONNEES**
- ✓ **MOCKING**
- ✓ **GÉNÉRATION DES CLIENTS**
- ✓ **SYNCHRONISATION DES CHANGEMENTS**



I won.

**EST-CE QUE JE DOIS TOUJOURS FAIRE DU
DESIGN FIRST ?**

ÇA DÉPEND™.

EN 2021, S'IL VOUS PLAÎT,
UTILISEZ DES CONTRATS
D'APIS 🙏

MERCI



Récupérez du swag en participant à notre recherche utilisateurs / sebastien@bump.sh

