APITHINKING

GROUPE ROCHER
reconnect people to nature

API Green Score

'API Numériquement Responsable'

API GREEN SCORE
by the API Thinking collective

Yannick Tremblais
IT Innovation Manager

ytremblais

GROUPE ROCHER
TECHNOLOGY SOLUTIONS

4 BUSINESS SECTORS

COSMETICS
CLOTHING
HOME CARE
WELL-BEING

GROUPE ROCHER

YVES ROCHER · Dr.Pierre Ricaud · arbonne. · PETIT BATEAU · STANHOME · KIOTIS PARIS · flormar · SABON

Present in nearly
120 countries

GROUPE ROCHER

reconnect people to nature

In 2019 Groupe Rocher adopt the **status of "mission-driven company"** .

Our **mission**:

Reconnect people to Nature

2030 GRTS IT RESPONSIBILITY AMBITIONS

GREEN CONQUEST 2030

GROUPE ROCHER

External Communities

Since 2020, the 1st French collective dedicated to APIS

A collective that draws its strength from its members

Free

Network

Diversity

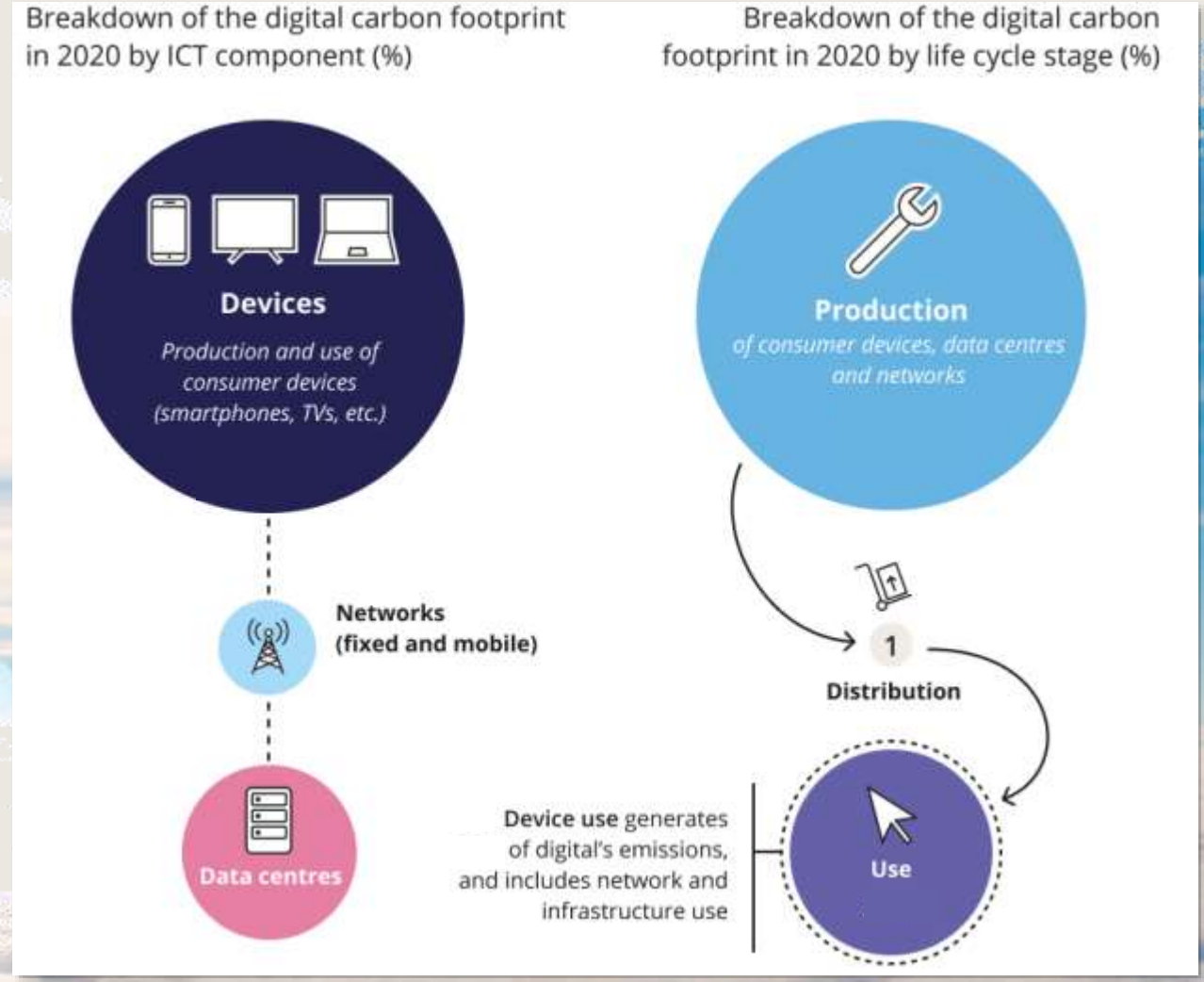
As we are pioneers on API Ecodesign

As environmental impact is a shared concern

As guideline and calculation rule must be shared and validated by many to be recognized

We are sharing our studies to external communities

Active Member of the API Thinking Community
head of workgroup “**Sustainable digital API**”



https://en.arcep.fr/uploads/tx_gspublication/press-kit-study-Ademe-Arcep-lot3_march2023.pdf

%	Energy	GHG	Water	Elec.	ADP
User equipment	60%	63%	83%	44%	75%
Network	23%	22%	9%	32%	16%
Data centres	17%	15%	7%	24%	8%

Breakdown of impact of the digital world in 2019

source: https://www.greenit.fr/wp-content/uploads/2019/11/GREENIT_EENM_etude_EN_accessible.pdf

https://pigrai.com/2022/11/03/ecologie-et-metaux-rares-aurore-stephant/

Tableau des Actions pour le Numérique Responsable Mise en Place des Actions

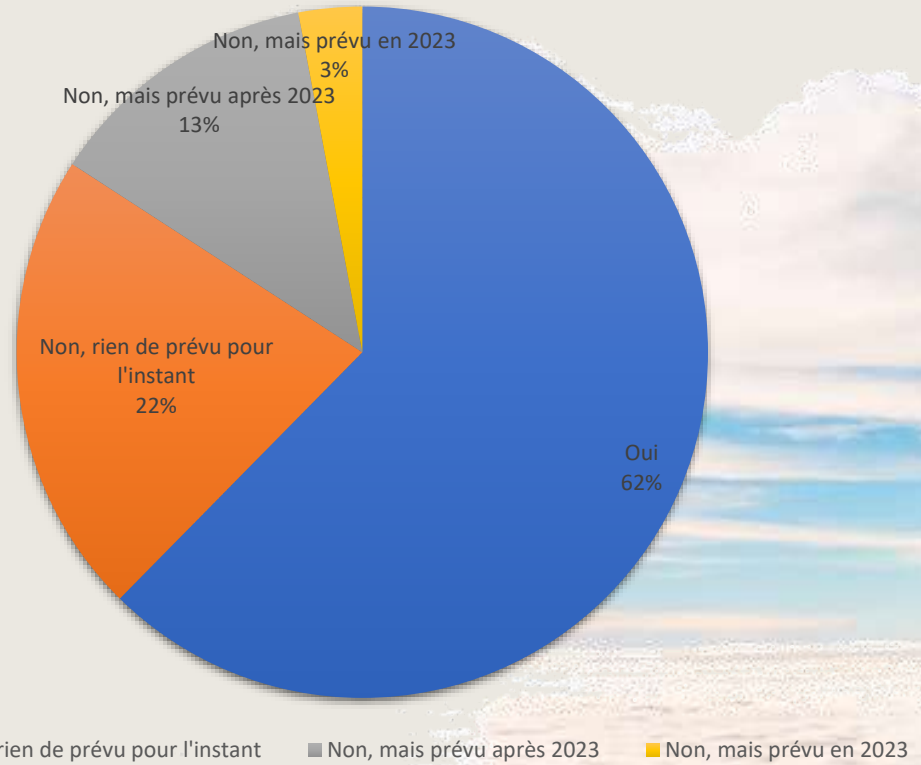
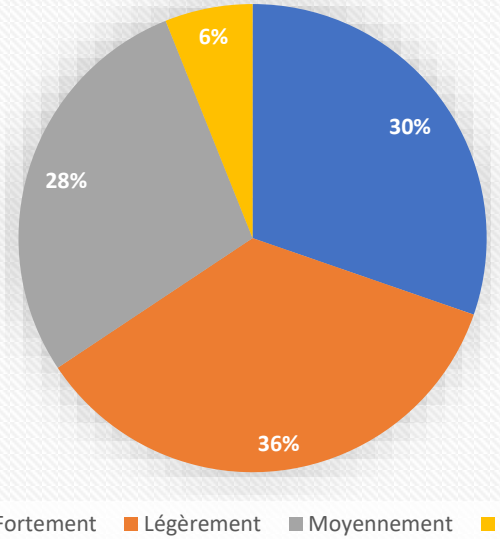
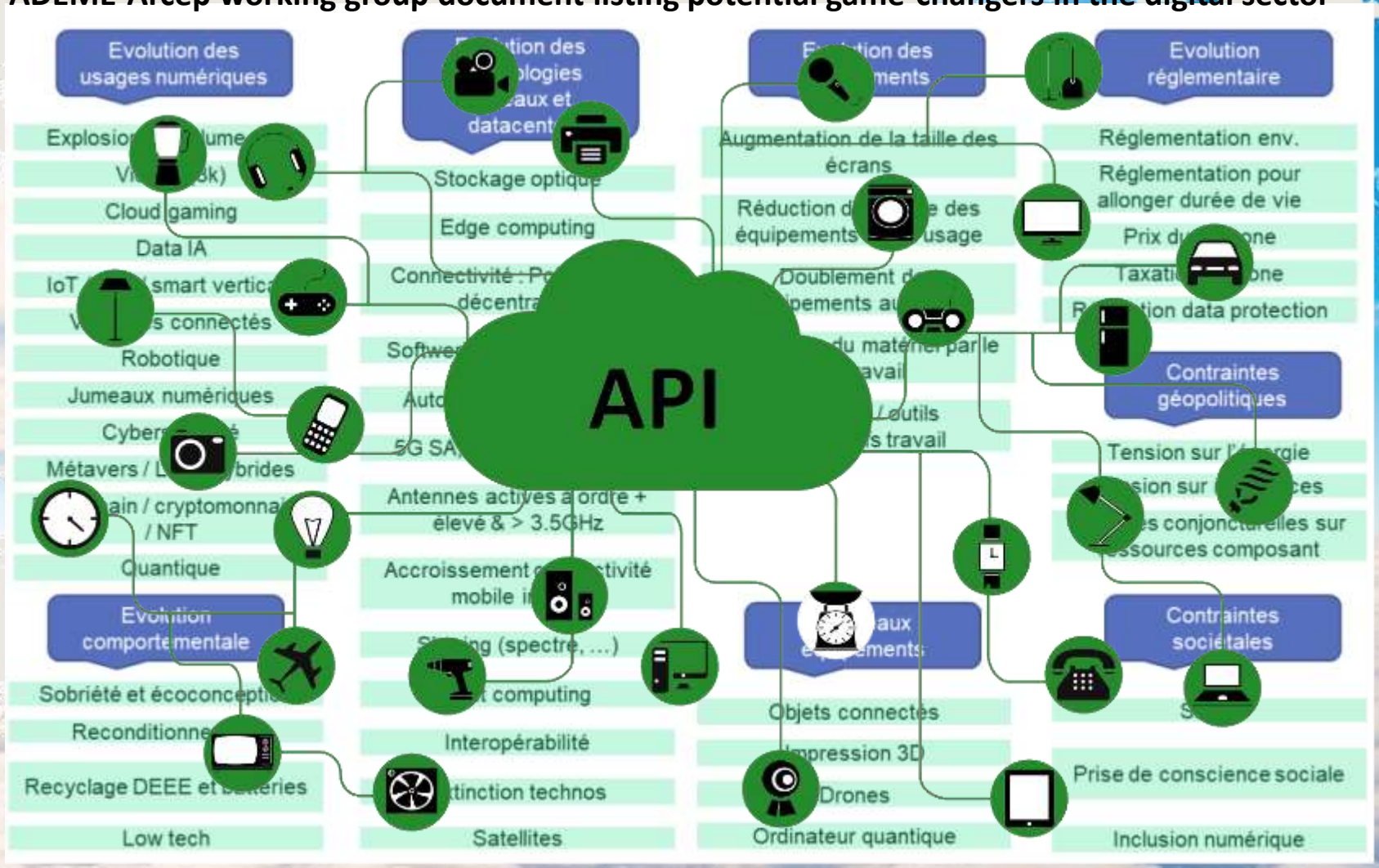


Tableau de l'Intensification des Actions en 2024



Without action to limit the growth of the environmental impact of digital technology, its carbon footprint could triple between 2020 and 2050

ADEME-Arcep working group document listing potential game-changers in the digital sector



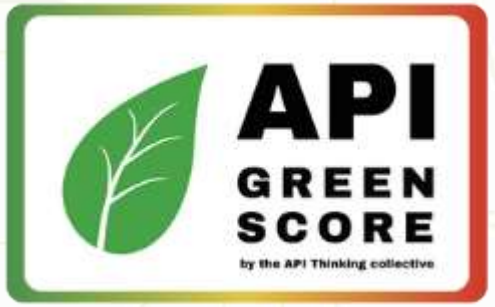
API Green Score

The API Green Score is a toolkit to help API consumers, designers and owners to ask themselves questions about the digital impact of their APIs

This tool is based on 7 different domain in order to create relevant and realistic metrics that stakeholders can use

The evaluation method is shared with all API Personas
(API owners, API consumers, API developers)

This toolkit concern eco-design and eco-consumption of API



Excellent	Acceptable	Average	Poor	Very Poor
A	B	C	D	E

7 Domains

API Lifecycle

- Decommission an unused API
- Deploy API near consumer
- Reduce number of API versions
- Unify API catalog
- Create consumer referential
- Identify API for single usage
- Urbanization with Data Governance

Data Exchange

- Exchange with Smallest Size
- Following API payload size
- Prefer Opaque Token to JWT
- API Customer Centricity principles
- API Data / Granularity
- Leverage Odata or GraphQL for DB APIs
- Data Management
- Dynamic Content

Data

- Optimize queries to limit returned information
- Collect only required data
- Provide only changed data
- Use cache
- Communicate on Payload size
- API used geolocally close to their consumers

Architecture

- Promote event architecture
- Filter data in payload
- Pagination
- Webhook or Business Notification
- AsyncAPI

Tools

- Define a basis of criteria for rating
- Provide KPIs (Nb of call, payload size, nb of equipment's used, ...)
- Evaluate energy consumption for one API
- Know language impact for energy consumption

Infrastructure

- Use adaptive infrastructure
- Use as few cloud suppliers as possible between consumer and backend
- Be near Data Center
- Define which actions are more relevant to do to reduce the impact of API ?

Communication

- Name of API Green Score
- Guideline resources
- Sharing criteria of evaluation and methods
- Adapt the communication of each personas

API Lifecyle Domain

API Lifecycle

API uses: (who, when, what)

Description

Have a consumer referential
What is the impact of this referential on the API Green Score?
Who consumes my API?
What: Which version of API?
When: Which number of asked calls vs number of calls ? Date of last call?
What is the calls volume ?

Governance

API Product Owner
Center of Expertise API

KPI per API

Nb of call per consumer
Nb of consumers per API
Nb of versions per API (US03)
Location of consumers
Documentation quality (US06)
What is the API Footprint?

Impact EcoScore

20%

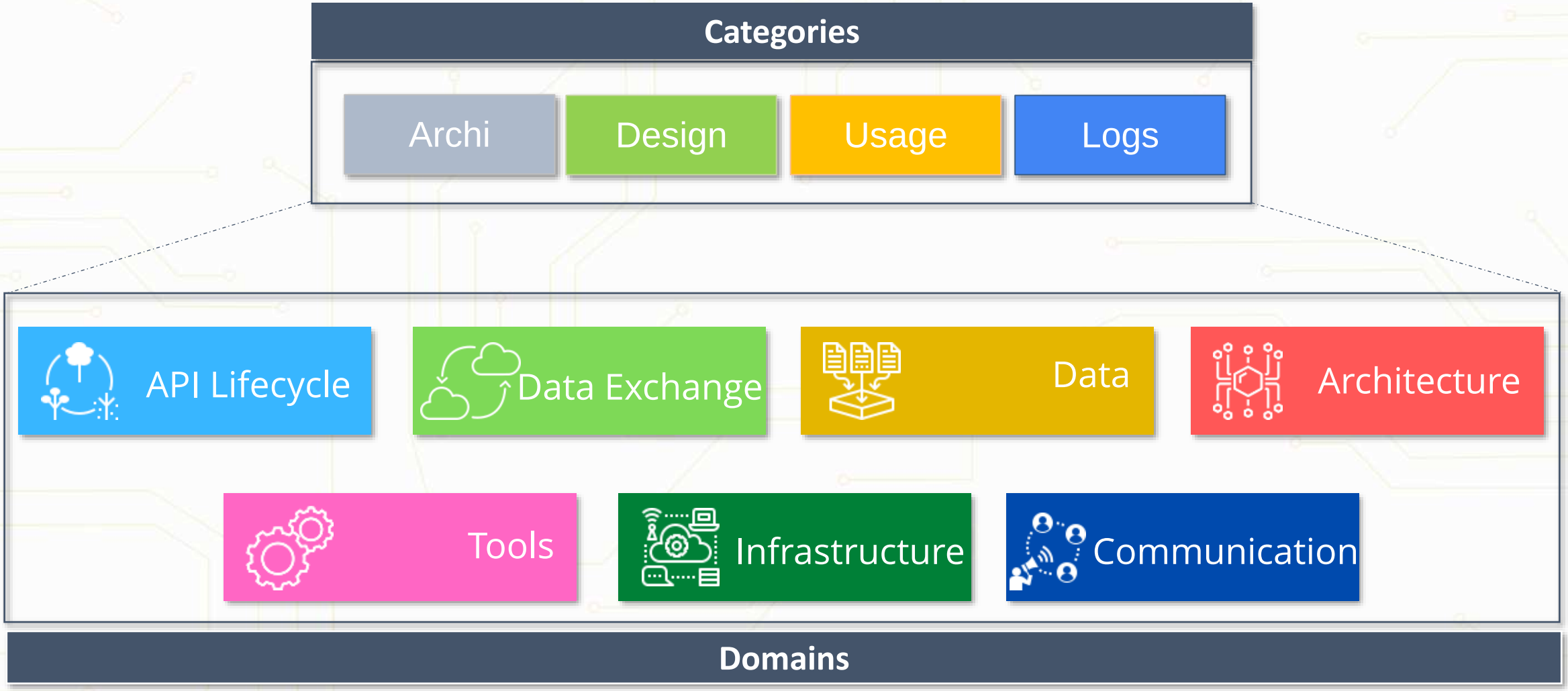
Tools to measure

Logs API / Operational Reporting
Analytics API Gateway
To influence the Metrics
API Gateway/API Portal

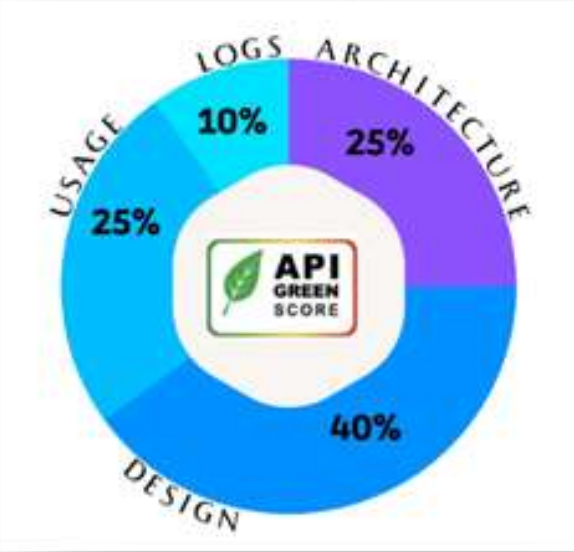
Example

API Order 10000/ call / month
API last Call
Nb of Consumers who used this API

Domains vs Categories



Evaluation Grid: Results



API : Green Score Grid

Section	RuleID	Items analysed	Description	Weight		Score Evaluation: Points	Total Weight	Eval	Score	Comment
Architecture	AR01	Use Event Driven Architecture to avoid polling madness and inform subscribers of an update	Use Event Driven Architecture to avoid polling madness.	25%	25,0%	375	6,25%	☐	0	
	AR02	API runtime close to the Consumer	Deploy the API near the consumer		25,0%	375	6,25%	☐	0	
	AR03	Ensure the same API does not exist *	Ensure only one API fit the same need		25,0%	375	6,25%	☐	0	
	AR04	Use scalable infrastructure to avoid over-provisioning	Use scalable infrastructure to avoid over-provisioning		25,0%	375	6,25%	☐	0	
Design	DE01	Choose an exchange format with the smallest size (JSON is smallest than XML)	Prefer an exchange format with the smallest size (JSON is smaller than XML).	40%	25,0%	600	10,00%	☐	0	
	DE02	new API --> cache usage	Use cache to avoid useless requests and preserve compute resources.		15,0%	360	6,00%	☐	0	
	DE03	Existing API --> cache usage efficiency	Use the cache efficiently to avoid useless resources consumtion.		20,0%	480	8,00%	☐	0	
	DE04	Opaque token usage	Prefer opaque token usage prior to JWT		2,0%	48	0,80%	☐	0	
	DE05	Align the cache refresh with the datasource **	Align cache refresh strategy with the data source		4,0%	96	1,60%	☐	0	
	DE06	Allow part refresh of cache	Allow a part cache refresh		4,0%	96	1,60%	☐	0	
	DE07	Is System, Business or cx API ?	Use Business & Cx APIs closer to the business need		10,0%	240	4,00%	☐	0	
	DE08	Possibility to filter results	Implement filtering mechanism to limit the payload size		2,5%	60	1,00%	☐	0	
	DE09	Leverage OData or GraphQL for your databases APIs	Leverage OData or GraphQL when relevant		10,0%	240	4,00%	☐	0	
	DE10	Redundant data information in the same API	Avoid redundant data information in the same API		5,0%	120	2,00%	☐		
	DE11	Possibility to fitler pagination results	Implement pagination mechanism to limit the payload size		2,5%	60	1,00%	☐	0	
Usage	US01	Use query parameters for GET Methods	Implement filters to limit which data are returned by the API (send just the data the consumer need).	25%	5,0%	75	1,25%	☐	0	
	US02	Decomission end of life or not used APIs	Decomission end of life or not used APIs		10,0%	150	2,50%	☐	0	
	US03	Number of API version <=2	Compute resources saved & Network impact reduced		10,0%	150	2,50%	☐	0	
	US04	Usage of Pagination of results available	Optimize queries to limit the information returned to what is strictly necessary.		10,0%	150	2,50%	☐	0	
	US05	Choosing relevant data representation (user don't need to do multiple calls) is Cx API ?	Choose the correct API based on use case to avoid requests on multiple systems or large number of requests. Refer to the data catalog to validate the data source.		20,0%	300	5,00%	☐	0	
	US06	Number of Consumers	Deploy an API well designed and documented to increase the reuse rate. Rate based on number of different consumers		25,0%	375	6,25%	0%	0	this a rate evaluation
	US07	Error rate	Monitor and decrease the error rate to avoid over processing		20,0%	300	5,00%	0%	0	this a rate evaluation
Logs	LO01	Logs retention	Align log retention period to the business need (ops and Legal)	10%	100,0%	600	10,00%	☐	0	
				100%		6000	100%		Rank 0	
									E	

Legend :

* > 70% redundant fields with on other API

** cache refresh must be equal to the data update frequency on the source system

Excellent	Acceptable	Average	Poor	Very Poor	Not evaluted
A	B	C	D	E	N.C
>=6000	6000<>=3000	3000<>=2000	2000<>=1000	<1000	

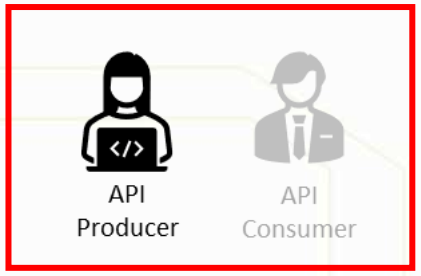
Architecture

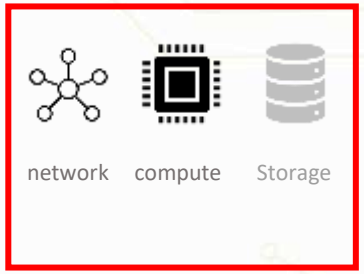
AR05: Carbon Footprint Dashboard

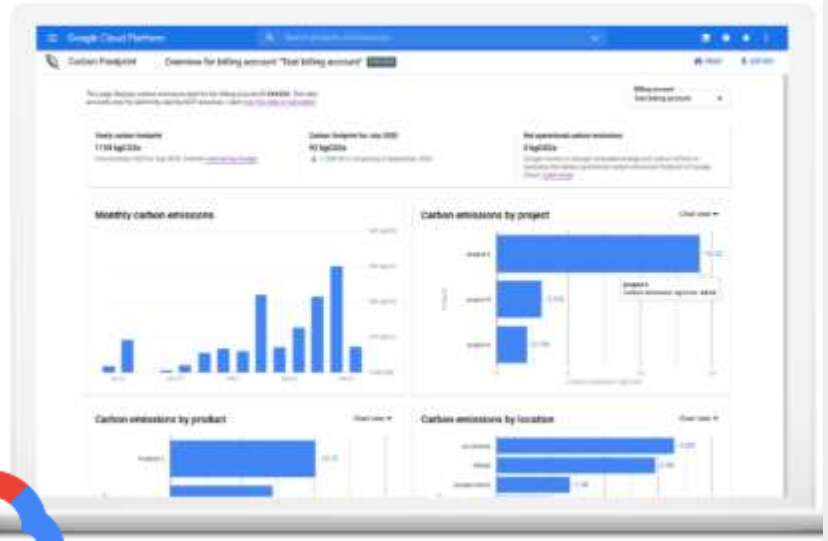
Some cloud providers produce carbon footprint dashboard.
You can implement your own or adapt it based on your infrastructure and be close to your usage.

Ex: evaluation the impact of compute, network, disk and divided by the number of calls of the evaluate API.

Expected gain: Network, compute







Project name:

Project label key

Project label value

1 Jan 2023 - 22 Nov 2023

Project overview

Cost & carbon emissions

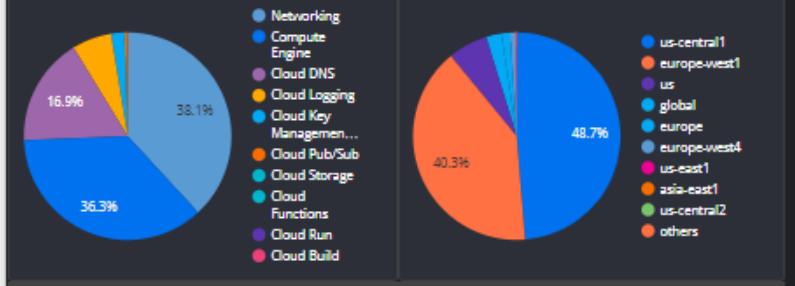
Total CO2 emissions equivalent (kg)

570.5

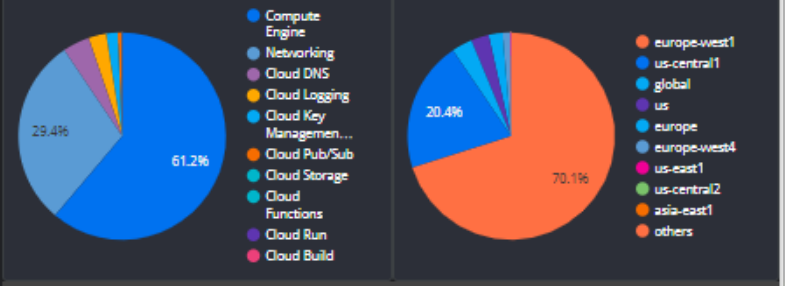
Total spent (EUR)

6.88K €

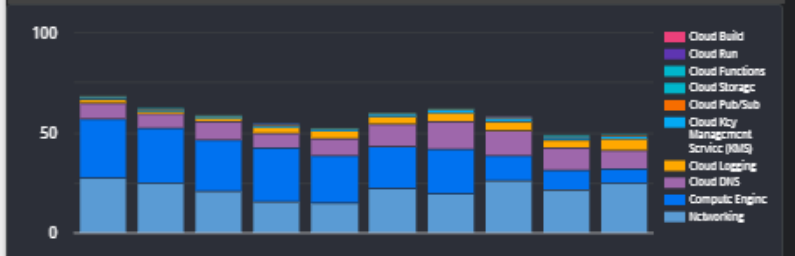
CO2 emissions equivalent breakdown per product and region



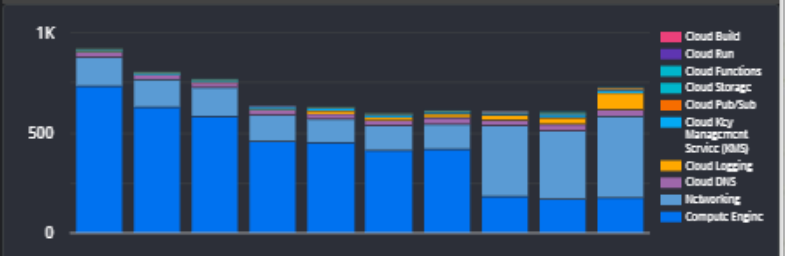
Spend breakdown per product and region



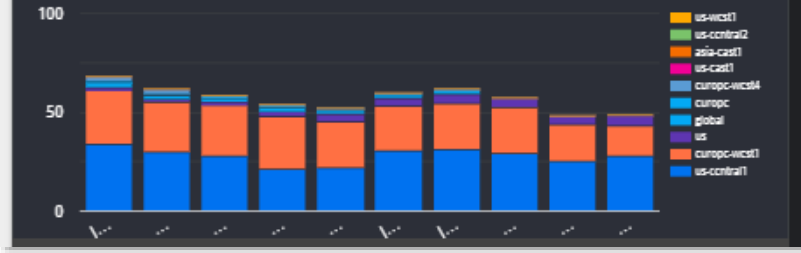
Breakdown of CO2 emissions equivalent per product (kg)



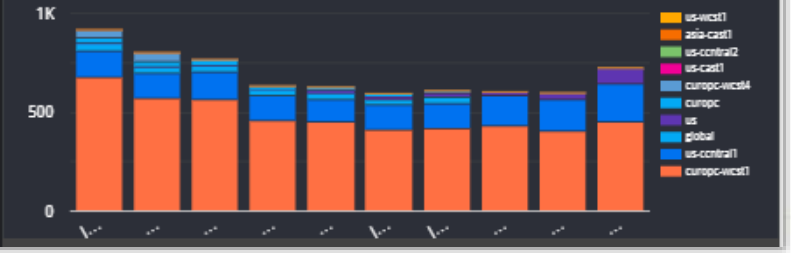
Spend breakdown per product (EUR)



Breakdown of CO2 emissions equivalent per region (kg)



Spend breakdown per region (EUR)



Data Exchange

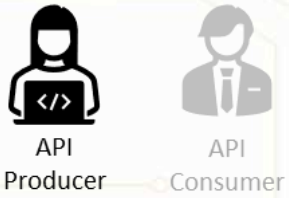
DE01: Prefer an exchange format with the smallest size (JSON is smaller than XML).

One of the structuring questions when designing an API is the selection of the exchange format to use. If the choice is often made by technical constraints or personal affinities, the durability aspect is also to be taken into account.

Indeed, there are exchange formats that are heavier than others. For example, between JSON is smaller than XML. The second format will therefore have a stronger impact on the network, the computing and the storage.

In the interest of sustainability, we recommend to use a lighter exchange format to reduce the bandwidth consumed for the requests, the compute and storage resources consumption used to process and store the payloads.

Expected gain: Network, compute and storage impact reduced



```xml
<?xml version="1.0" ?>
<order>
  <customer>
    <lastName>Smith</lastName>
    <firstName>John</firstName>
    <email>john.smith@example.com</email>
    <phone>123-456-7890</phone>
  </customer>
  <billingAddress>
    <street>123 Liberty Street</street>
    <city>New York</city>
    <postalCode>10001</postalCode>
    <country>USA</country>
  </billingAddress>
  <shippingAddress>
    <street>456 Freedom Road</street>
    <city>New York</city>
    <postalCode>10002</postalCode>
    <country>USA</country>
  </shippingAddress>
  ...
  <paymentMethod>Credit Card</paymentMethod>
  <deliveryMethod>Standard</deliveryMethod>
</order>
```

880 bytes bookstore.xml

```json
{
    "customer": {
        "lastName": "Smith",
        "firstName": "John",
        "email": "john.smith@example.com",
        "phone": "123-456-7890"
    },
    "billingAddress": {
        "street": "123 Liberty Street",
        "city": "New York",
        "postalCode": "10001",
        "country": "USA"
    },
    "shippingAddress": {
        "street": "456 Freedom Road",
        "city": "New York",
        "postalCode": "10002",
        "country": "USA"
    },
    ...
    "paymentMethod": "Credit Card",
    "deliveryMethod": "Standard"
}
```

988 bytes bookstore.json

		Xml	json	Gain
		880 B	988 B	108 B (12%)
nb call /day	2 000 000	1,64 GB	1,84 GB	0,2 MB
nb call/month *(30)*	60 000 000	49,17 GB	55,21 GB	**6,04 GB**

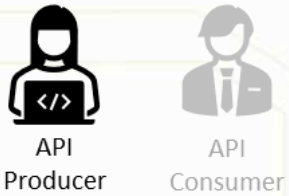
DE02/DE03/DE05: Use cache to avoid useless requests and preserve compute resources.

The use of a cache has become common in computer architectures to store frequently used information on a fast storage.

In addition to improving the response time of APIs, and therefore the consumer's experience of the service, it also saves computational resources by avoiding executing the same query on the same data multiple times.

It is recommended to place a cache in front of each brick of an architecture returning data (API, database, frontend application, ...) and close to the users to preserve compute resources and improve performances of the API.

Expected gain: Compute resources saved & Network impact reduced



L'ORÉAL IT

API center of enablement

- Home
- API satisfaction
- Performance
- Quality Review
- Business API
- APIs Usage
- Consumers

Support

ccoe-report-support

api-support@loreal.com

API Proxies : global-product-elixpedia-skuget

TimeRange (utc) : 15/07/2020 — 25/11/2022

Latency Year *

8,49

* Average Latency (**s**)

2022

Availability Year *

94,07 %

*Average current year

2022

Capacity to depreciate older versions *

95,25 %

# of calls on API last version / # of calls all version

Latency Month *

7,57

*Average Latency (**s**)

11/2022

Availability Month*

87,15 %

* Request without error 5xx

11/2022

Caching Performance

The number of successful API requests that use the Response Cache **VS** The total number of API calls

Exposed API OAUTH2.0 authorization *

100,00 %

Frontend

0,00 %

Backend

# of API with oauth2.0 authorization / total # API

Latency Over Time (second)

Availability Over Time (%)

Error Type Over Time (%)

Data Exchange

US01 : Use query parameters for GET Methods

Optimize queries to limit the information returned to what is strictly necessary.

It is often observed that requests made on APIs are not precise enough, which returns a volume of information greater than necessary.

This results in increased bandwidth consumption during exchanges.

The best practice is to create precise requests that return, as much as possible, the strictly necessary information, thus avoiding the transfer of useless information.

This rule is linked to DE08 : "Implement filters to limit which fields are returned by the API "

Expected gain: Network, compute

https://api-adresse.data.gouv.fr/search

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

adresse.data.gouv.fr
Le site national de l'adresse

```
curl "https://api-adresse.data.gouv.fr/search/?q=10+av"
```

https://api-adresse.data.gouv.fr/search/?q=10+av

5 results

Payload : 2,57 KB

https://api-adresse.data.gouv.fr/search/?q=10+av&limit=100

100 results

Payload 42,9 KB

https://api-adresse.data.gouv.fr/search/?q=10+av&limit=1000


```
{
    "code": 400,
    "message": "limit must be an integer between 1 and 100"
}
```


```
curl "https://api.spotify.com/v1/browse/categories"
```

https://api.spotify.com/v1/browse/categories?offset=0&limit=50

48 results

Payload : 16,37 KB

https://api.spotify.com/v1/browse/categories?offset=30&limit=50

18 results

Payload 6,81 KB

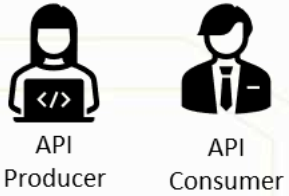
Tools

LO01: Collect only required data and use the right retention time according to the business requirements.

It is quite common for applications to store a large amount of useless information without time limit.
This results in an excessive consumption of storage services for data that will not be used or no longer used.

It is necessary to clean up the data in order to keep only the data that is useful and to define a coherent retention policy in order to delete them once their validity or exploitation period has passed.

Expected gain: Volume of data stored reduced & Network impact reduced



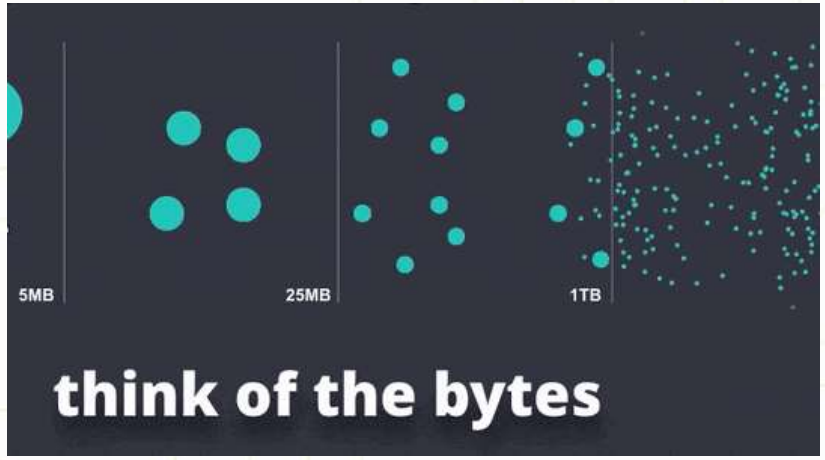
Time	Brand	Country	Method	Global Response Time
Dec 1, 2022 @ 17:44:36.281	YR	FR	GET	60
Dec 1, 2022 @ 17:44:36.081	YR	BE	POST	336
Dec 1, 2022 @ 17:44:35.880	YR	BE	POST	40
Dec 1, 2022 @ 17:44:35.870	YR	FR	GET	396
Dec 1, 2022 @ 17:44:35.880	YR	US	POST	150
Dec 1, 2022 @ 17:54:22.130	YR	BE	POST	500
Dec 1, 2022 @ 17:44:36.810	YR	US	POST	200

detail by each API call

automatic rollup process
to reduce granularity

Time	Brand	Country	Method	Global Response Time	Nb of Call
Dec 1, 2022 @ 17:00	YR	FR	GET	228	2
Dec 1, 2022 @ 17:00	YR	BE	POST	292	3
Dec 1, 2022 @ 17:00	YR	US	POST	175	2

detail by each hour call
keep details by brand/country/method
aggregate : Average(Response Time), sum (nb of call) by hour



Major Impact: save disk space

https://www.elastic.co/guide/en/elasticsearch/reference/current/rollup-overview.html

- Define log retention per services/domain
- Define log rotation (rollup)
- Keep unit date for 45 days, and aggregate them(mn/hour)

➔ Reduce size of Elastic indices to 4Go /month

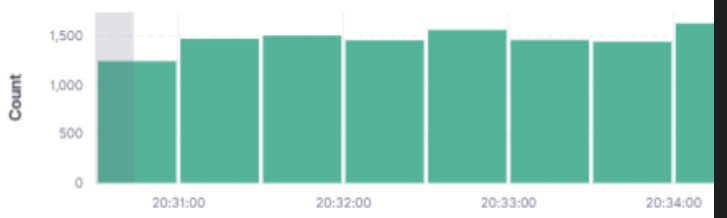
main.yml 441 Bytes

```yaml
---
envt: prd

hypervision_retention: "45"
hypervision_retention_gateway_log: "180"

```

omni_rollup-middlewares-api-apim-by-day.json 750 bytes

```json
{
  "index_pattern":"middlewares-api-apim-*",
  "rollup_index":"rollup-middlewares-api-apim-by-day",
  "cron":"0 0 0 * * ?",
  "page_size": 1000,
  "groups":{
    "date_histogram":{
      "field":"@timestamp",
      "fixed_interval":"24h"
    },
    "terms":{
      "fields":[
        "gr-commercialnetwork.keyword",
        "gr-application.keyword",
        "gr-country.keyword",
        "gr-brand.keyword",
        "serviceName.keyword",
        "service_completed.keyword",
        "method.keyword",
        "hypervision_env.keyword"
      ]
    }
  },
  "metrics":[
    {
      "field":"global_response_time",
      "metrics":[
        "avg"
      ]
    },
    {
      "field":"routing_time",
      "metrics":[
        "avg"
      ]
    }
  ]
}
```

46,361 hits



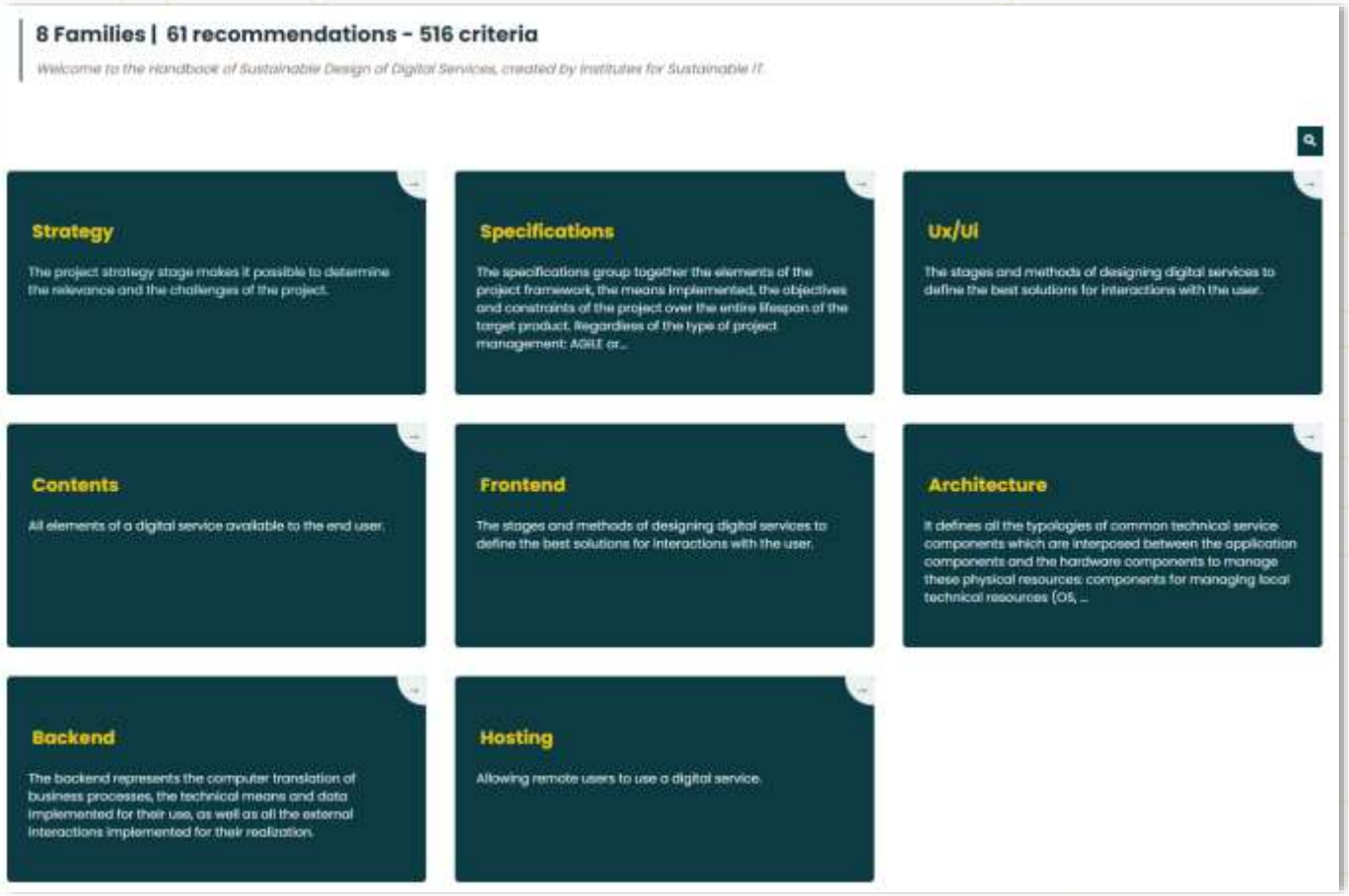
Time	gr-brand	gr-country	hypervision_retention	response_size
Jun 30, 2021 @ 20:45:39.997	YR	FR	45	30
Jun 30, 2021 @ 20:45:39.992	YR	FR	45	236
Jun 30, 2021 @ 20:45:39.966	YR	BE	45	0
Jun 30, 2021 @ 20:45:39.962	YR	BE	45	7,158

Institut du Numérique Responsable

TAKE AWAY

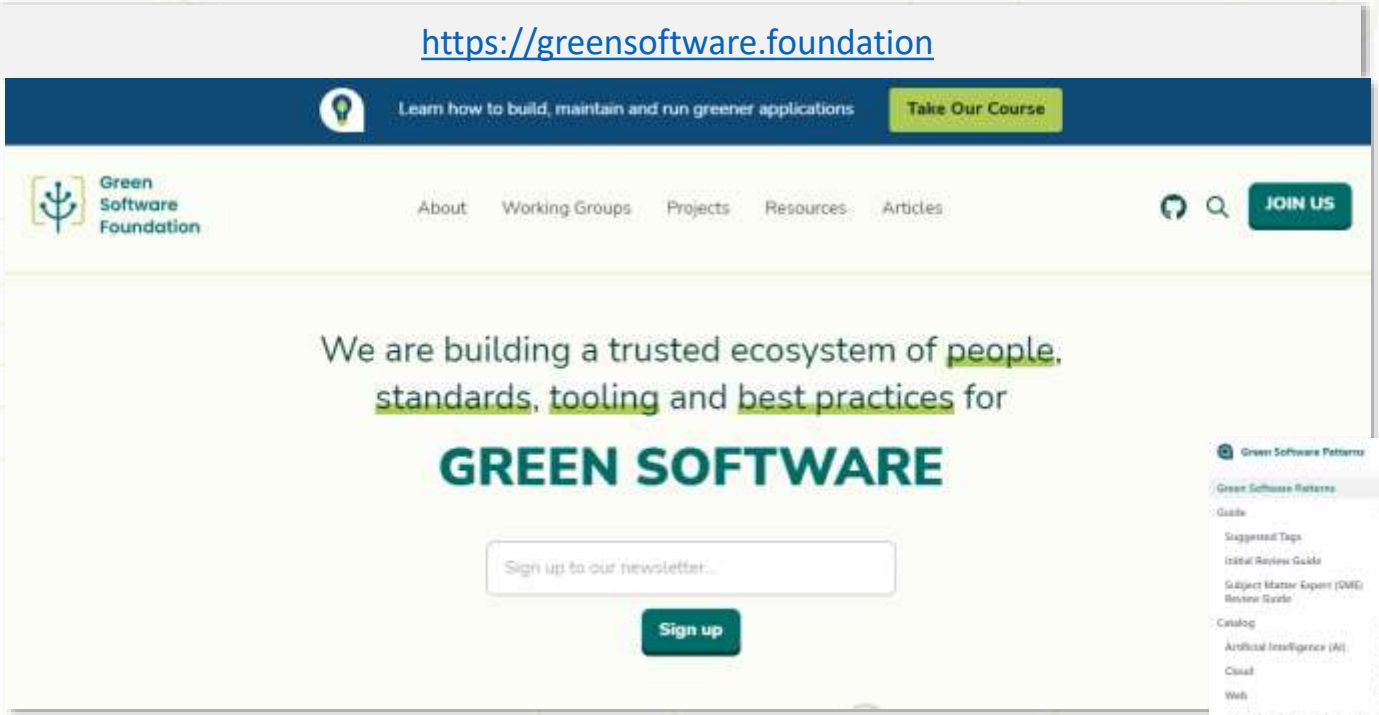
https://institutnr.org/guide-de-reference-de-conception-responsable

https://gr491.isit-europe.org/en

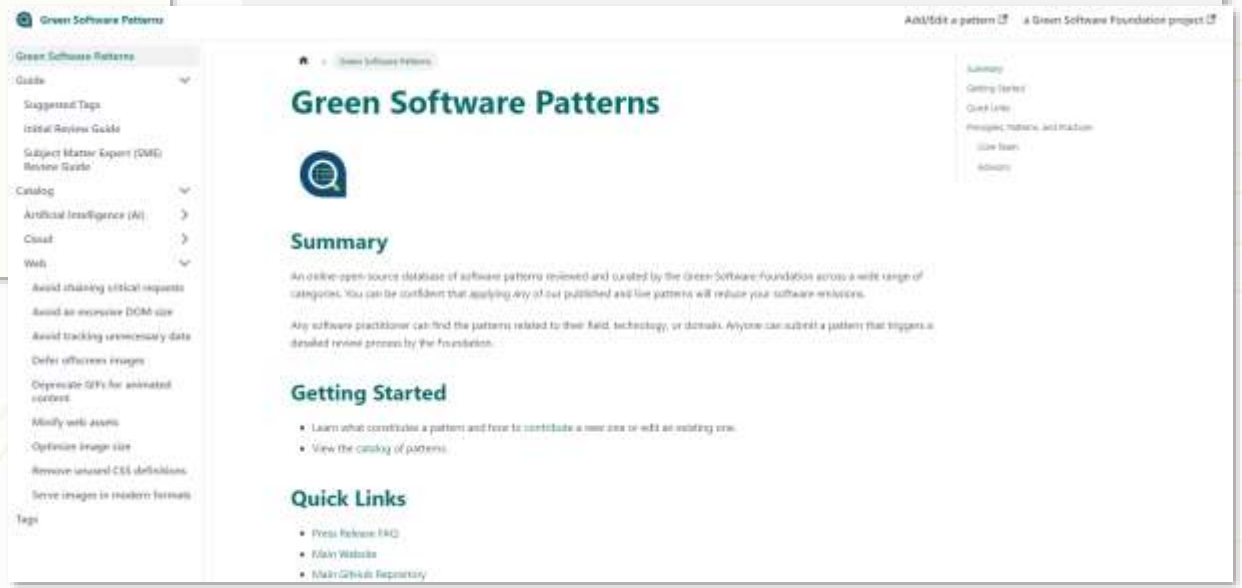


Green Software Foundation

https://greensoftware.foundation



https://patterns.greensoftware.foundation

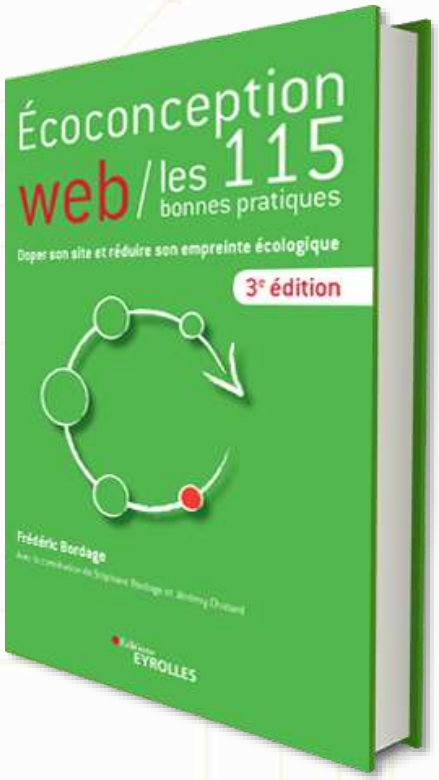


Collectif Numérique Responsable (CNUMR)

TAKE AWAY

https://collectif.greenit.fr

https://github.com/cnumr



Collectif Conception Numérique Responsable

150 followers · France · https://collectif.greenit.fr · @cnumr · @greenit

Popular repositories

best-practices — Public
115 Web Ecodesign Best Practices
☆ 246 · 34

GreenIT-Analysis — Public
GreenIT-Analysis
JavaScript · ☆ 126 · 29

ecoCode — Public archive
Reduce the environmental footprint of your software applications with this cutting-edge sonarQube plugin
Groovy · ☆ 63 · 49

EcoIndex — Public
Dépôt du site www.ecoindex.fr
JavaScript · ☆ 44 · 9

GreenIT-Analysis-cli — Public
Wrapper de l'extension GreenIT Analysis
JavaScript · ☆ 33 · 15

ecoindex_cli — Public
This tool provides an easy way to analyze websites with Ecoindex from your local computer. You have the ability to make the analysis on multiple pages with multiple screen resolution. You can also ...
Python · ☆ 31 · 2

EcoCode

https://www.ecocode.io

11 Decembre 2020

Olivier Le Goaer

Android Java

Uncompressed Data Transmission

Transmitting a file over a network infrastructure without compressing it consumes more energy than with compression. More precisely, energy efficiency is improved in case the data is compressed at least by 10%, transmitted and decompressed at the other network node. From the Android client side, it means making a post HTTP request using a GZIPOutputStream instead of the classical OutputStream, along with the HttpURLConnection object.

ecoCode : comment réduire l'empreinte carbone de vos projets en utilisant des outils open-source

Green — *Amphi C*

L'empreinte carbone des services numériques est en constante augmentation, ce qui entraîne des conséquences négatives sur notre environnement. Dans cette conférence, nous explorerons ecoCode, un projet open-source qui vise à réduire l'empreinte carbone des projets de développement en proposant des outils et des pratiques durables. Nous présenterons l'étendu de cet écosystème de plugins mobiles et web multilangages SonarQube et vous montrerons comment les utiliser pour mesurer l'empreinte carbone de votre code et réduire votre impact environnemental. Pour comprendre la mise en place et l'emploi des plugins ecoCode android (mobile) et Python (web) : rien de tel qu'un live coding !

Speakers : David De Carvalho, Johanna Duigou

TAKE AWAY

https://github.com/green-code-initiative

Pinned

ecoCode Public

Reduce the environmental footprint of your software programs with SonarQube

Java ☆ 52 41

ecoCode-mobile Public

Reduce the environmental footprint of your mobile apps with SonarQube

Groovy ☆ 52 15

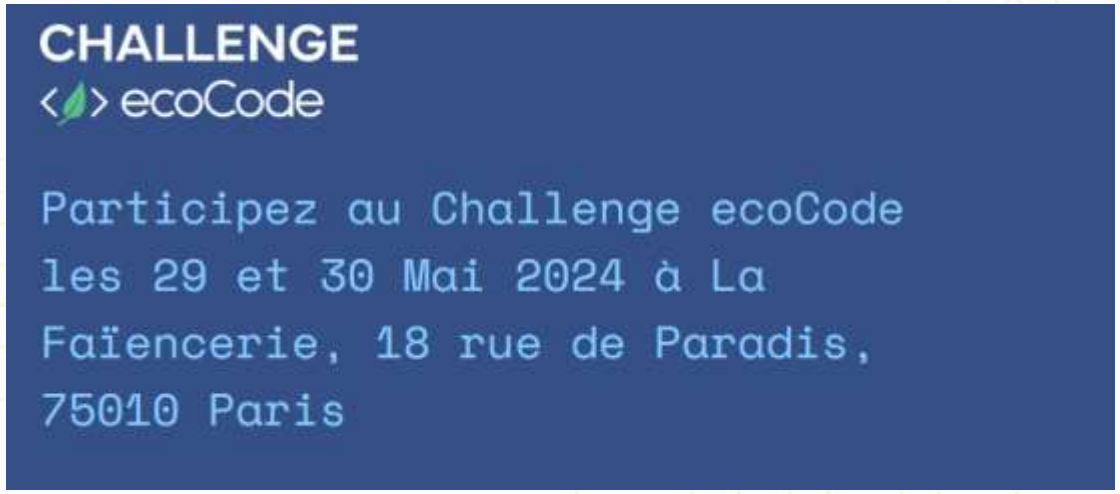
ecoCode-ios Public

Reduce the environmental footprint of your iOs mobile apps with SonarQube

Java ☆ 2

Challenge EcoCode

https://challenge.ecocode.io



TAKE AWAY

<3 OBJECTIFS>

Open source	Participez à l'enrichissement du plugin SonarQube en contribuant technologiquement avec l'ensemble des développeuses / développeurs.
Green code	La mission du collectif ecoCode est d'outiller les développeuses / développeurs pour faciliter la mise en œuvre de pratiques d'eco-coding.
Collaboration	La force de l'intelligence collective au profit de la co-construction d'outils pour un numérique écoconçu.

EcoCode

https://www.ecocode.io

11 Decembre 2020

Olivier Le Goaer

Android Java

Uncompressed Data Transmission

Transmitting a file over a network infrastructure without compressing it consumes more energy than with compression. More precisely, energy efficiency is improved in case the data is compressed at least by 10%, transmitted and decompressed at the other network node. From the Android client side, it means making a post HTTP request using a GZIPOutputStream instead of the classical OutputStream, along with the HttpURLConnection object.

ecoCode : comment réduire l'empreinte carbone de vos projets en utilisant des outils open-source

Green — *Amphi C*

L'empreinte carbone des services numériques est en constante augmentation, ce qui entraîne des conséquences négatives sur notre environnement. Dans cette conférence, nous explorerons ecoCode, un projet open-source qui vise à réduire l'empreinte carbone des projets de développement en proposant des outils et des pratiques durables. Nous présenterons l'étendu de cet écosystème de plugins mobiles et web multilangages SonarQube et vous montrerons comment les utiliser pour mesurer l'empreinte carbone de votre code et réduire votre impact environnemental. Pour comprendre la mise en place et l'emploi des plugins ecoCode android (mobile) et Python (web) : rien de tel qu'un live coding !

Speakers : David De Carvalho, Johanna Duigou

TAKE AWAY

https://github.com/green-code-initiative

Pinned

ecoCode Public

Reduce the environmental footprint of your software programs with SonarQube

Java ☆ 52 41

ecoCode-mobile Public

Reduce the environmental footprint of your mobile apps with SonarQube

Groovy ☆ 52 15

ecoCode-ios Public

Reduce the environmental footprint of your iOs mobile apps with SonarQube

Java ☆ 2

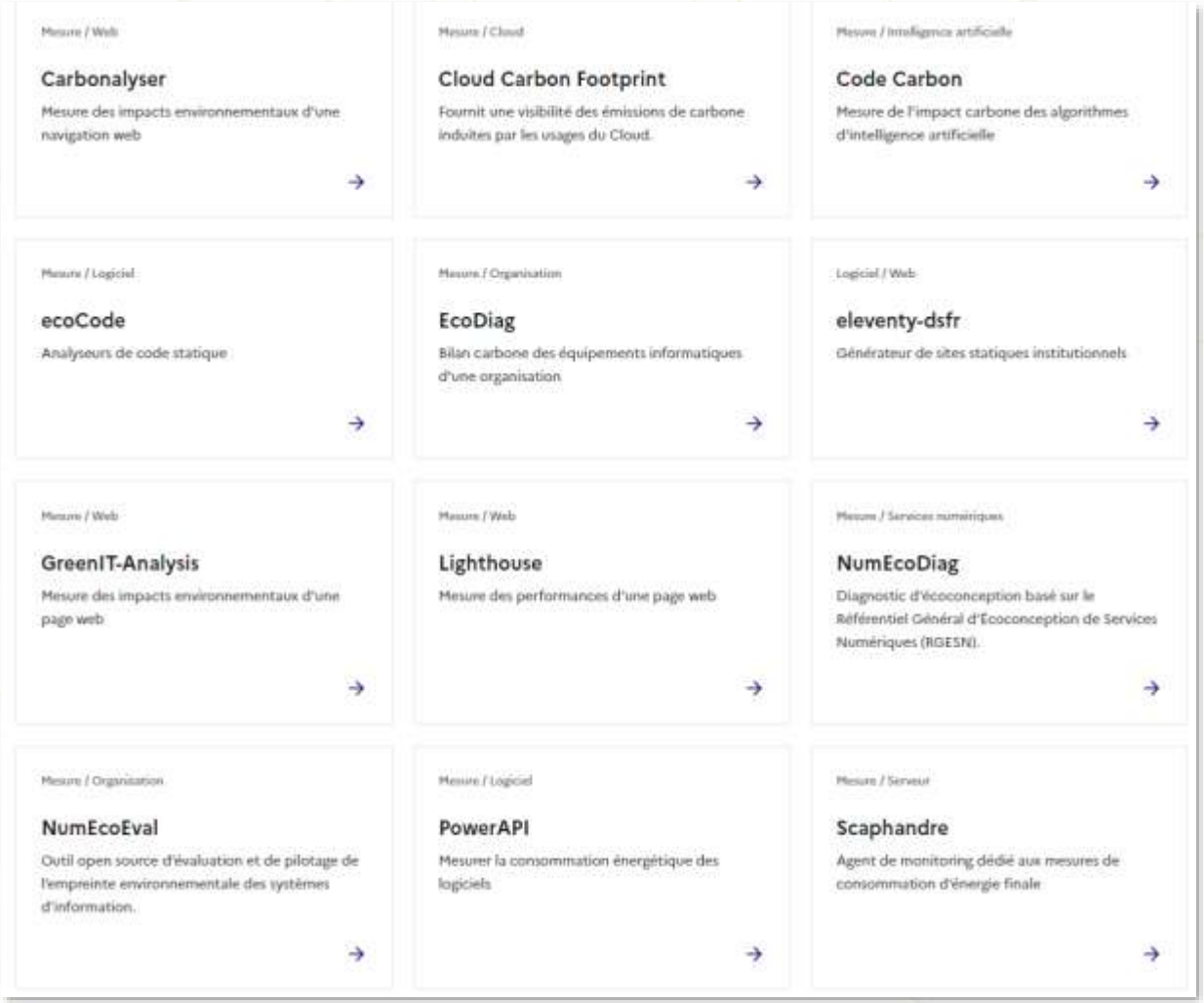
RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

mi num_ eco

Mission interministérielle
Numérique écoresponsable

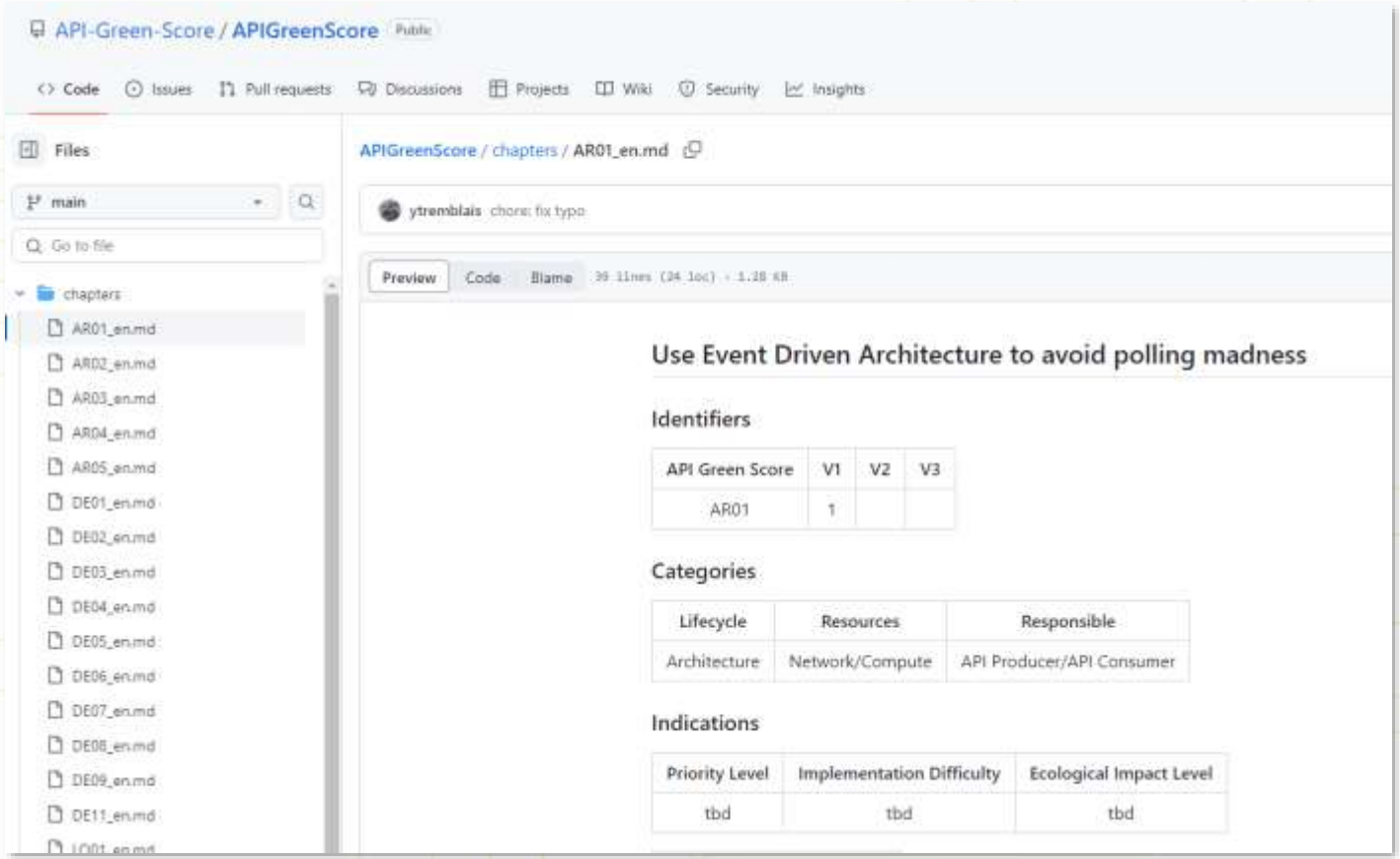
TAKE AWAY

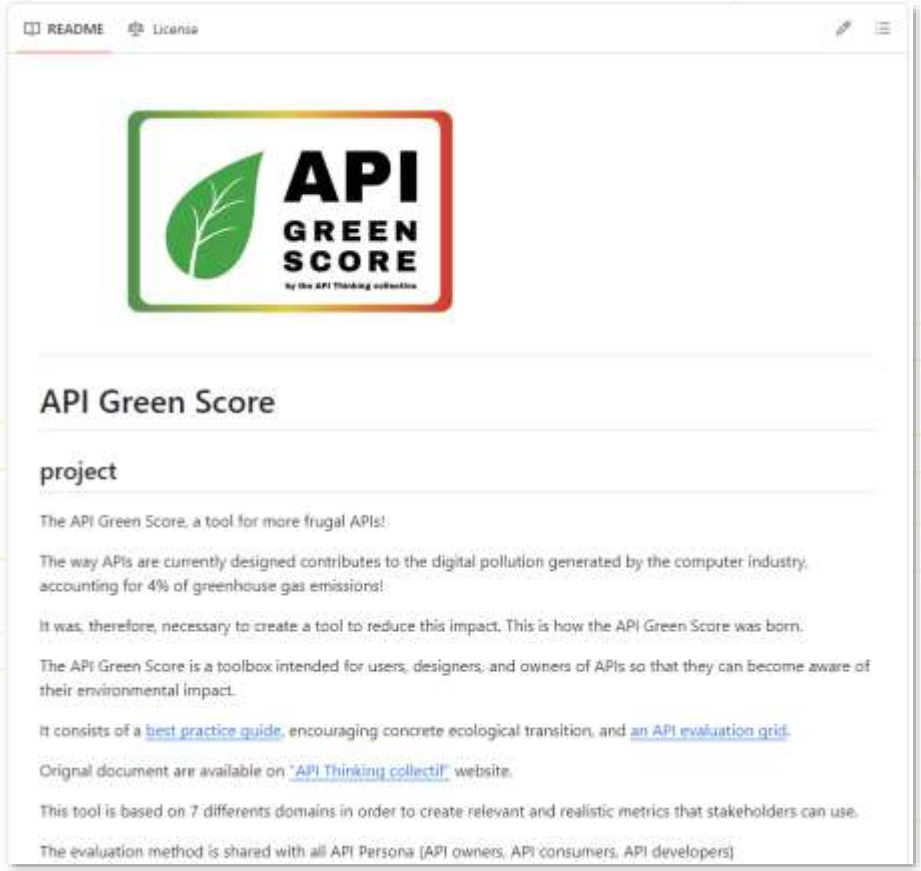
https://ecoresponsable.numerique.gouv.fr/publications/boite-outils



API Green Score Github

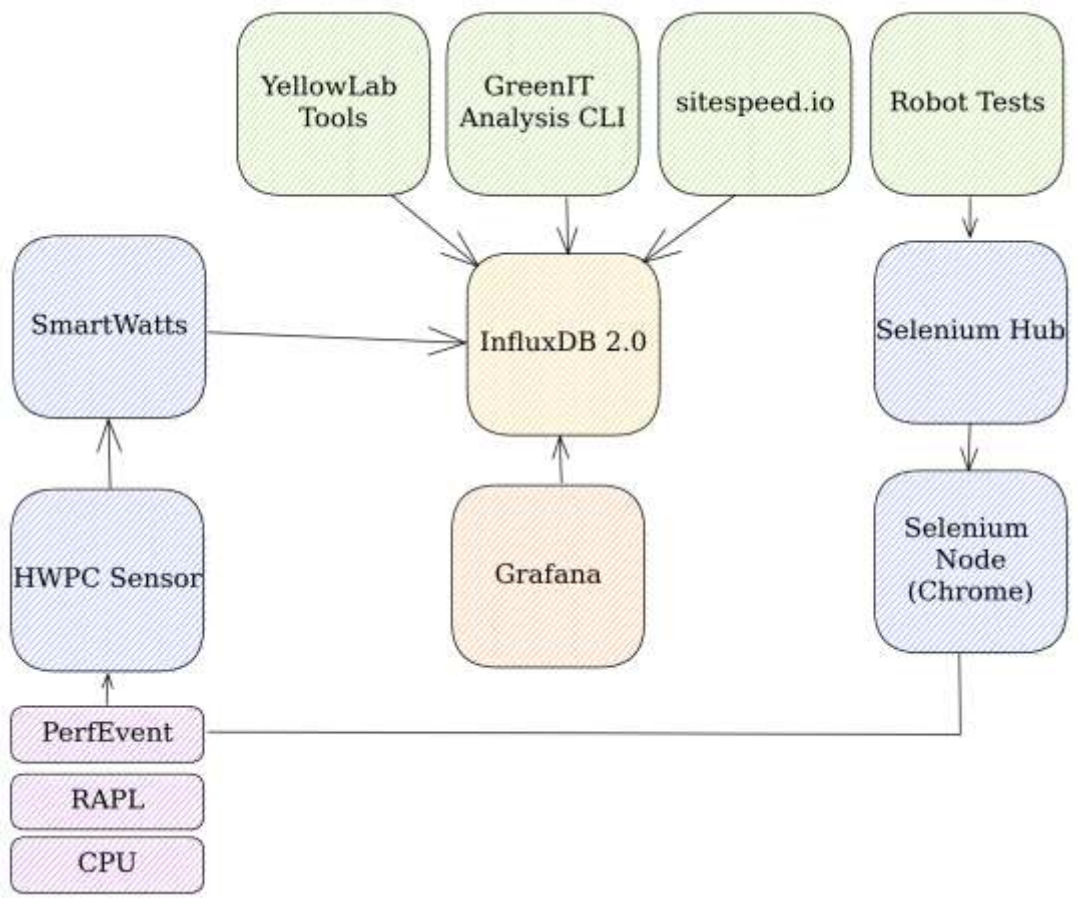
https://github.com/API-Green-Score/APIGreenScore

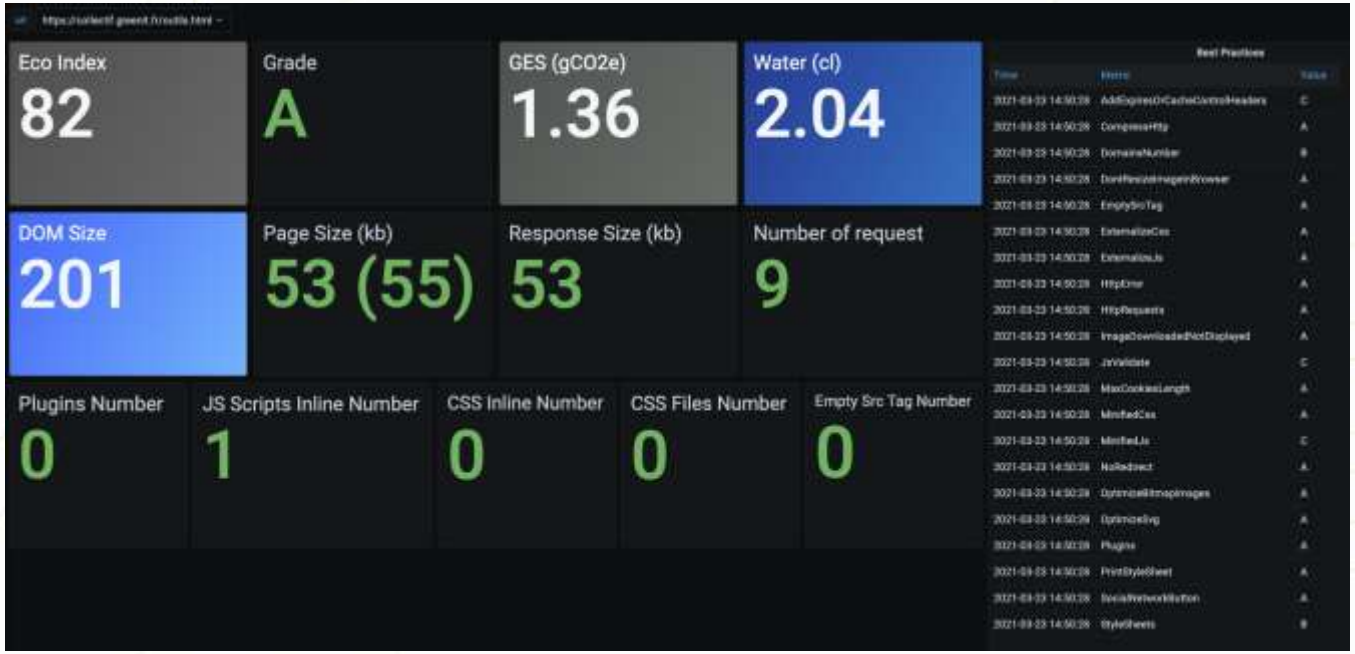




Tools

https://github.com/Zenika/pagiel





This is the beginning of our journey; we need you to contribute!

"Transform your API strategy with the industry's leading practices for environmental sustainability."

APITHINKING

https://www.collectif-api-thinking.com