Good API Product Lifecycle

Adam Kliment
a@goodapi.co

Adam Kliment

ORACLE® + apiary

Microsoft   salesforce   Bloomberg   VIACOM   xero

DigitalGlobe   GoPro   NBC UNIVERSAL

301,122 ARE WORKING ON **401,023** BEING CONSUMED BY **3M+**
HEROES   APIS   DEVELOPERS

goodapi.co

GOOD API

Independent API Consulting

Adam Kliment
Follow @ntmlk

Erik Wilde
Follow @dret

Zdenek "Z" Nemec
Follow @zdne

adidas

angelcam

Deutsche Post DHL Group

LEI GLOBAL LEGAL ENTITY IDENTIFIER FOUNDATION

KONICA MINOLTA

mystay

WINDING TREE
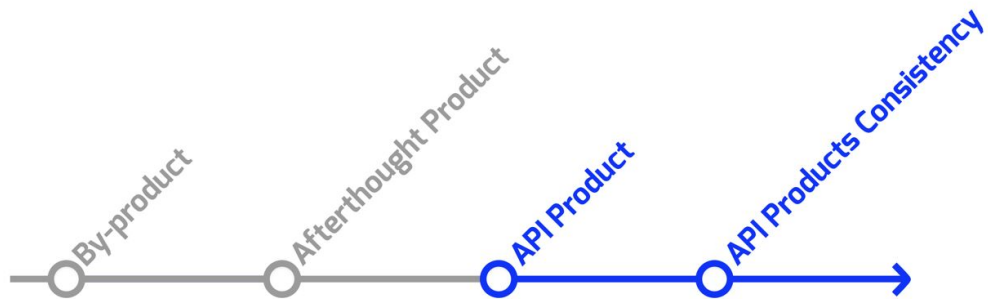
ytica

Treat API Development as Product Development
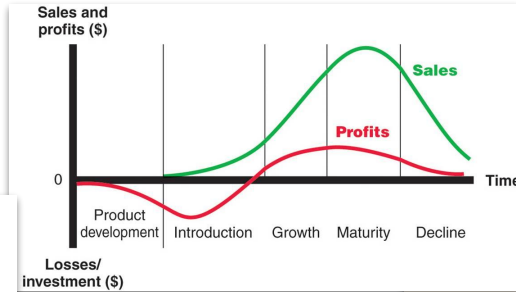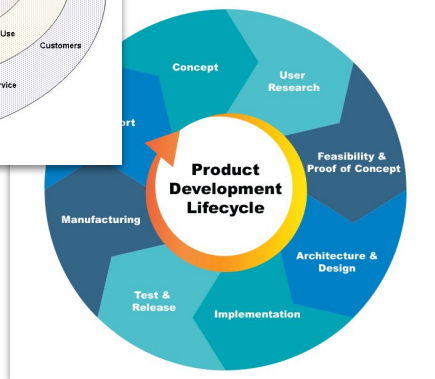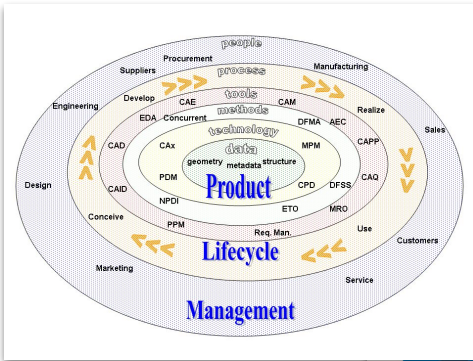
The API Product

History of API Products

The API Product

API-as-a-Product

Direct API monetization.

Where possible, the company product is offered directly through a paid API.

Product as API

Indirect "monetization".

Each product is an API and used through its API. APIs are contributing indirectly to the bottomline through increasing stickiness, driving revenue-generating activities, leveraging partner channels, increasing efficiency or leading innovation.
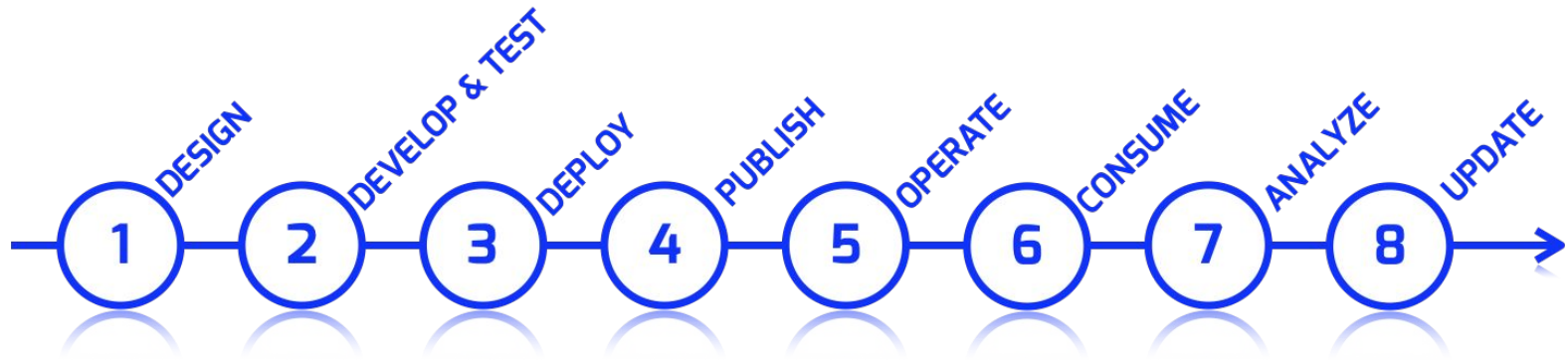
API Lifecycle

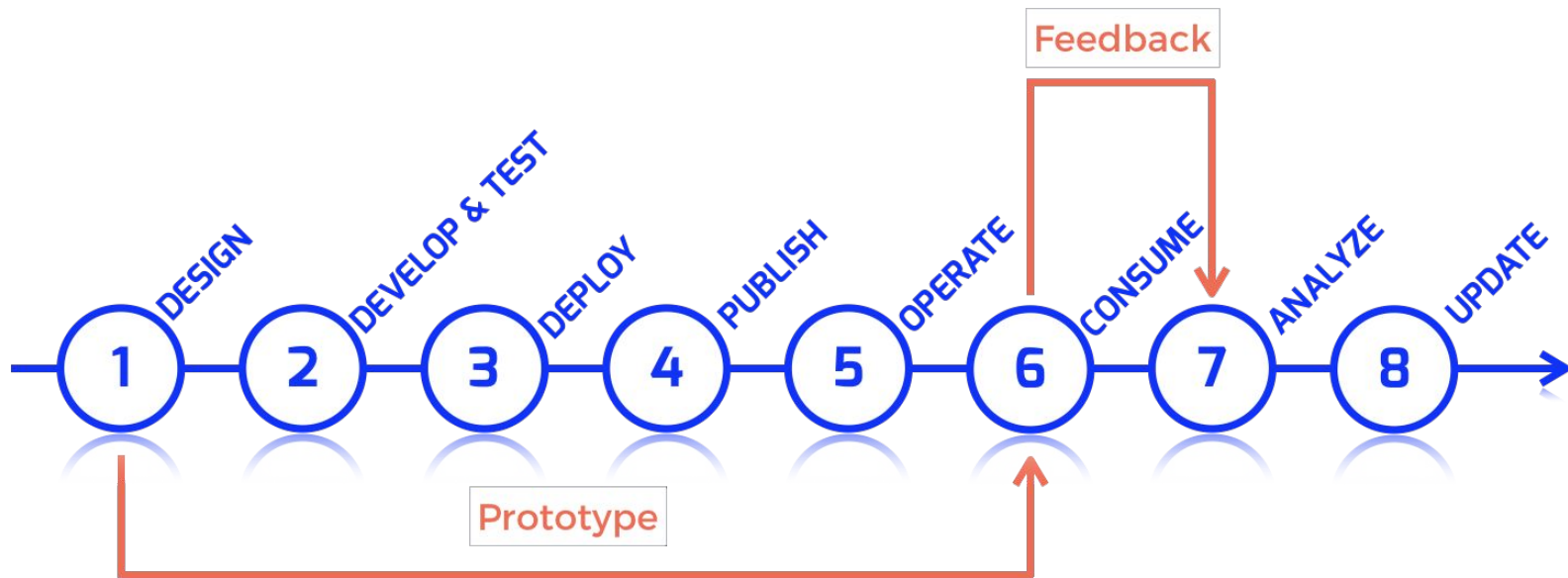# Waterfall Lifecycle

Agile API Lifecycle

Feedback

DESIGN
DEVELOP & TEST
DEPLOY
PUBLISH
OPERATE
CONSUME
ANALYZE
UPDATE

1 2 3 4 5 6 7 8

Prototype

http://www.company.com/index.htm

LOGO | COMPANY NAME

Search

Headline

Subhead 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus tincidunt suscipit orci hendrerit malesuada id et tellus. Ut rutrum quam eget enim ultricies sodales. Quisque efficitur erat a volutpat tristique.

Button

Subhead 1

Fusce condimentum purus in urna scelerisque volutpat. Duis porttitor massa nec enim malesuada, vitae aliquam neque vulputate. Morbi dictum ex porta, condimentum diam et, placerat magna.
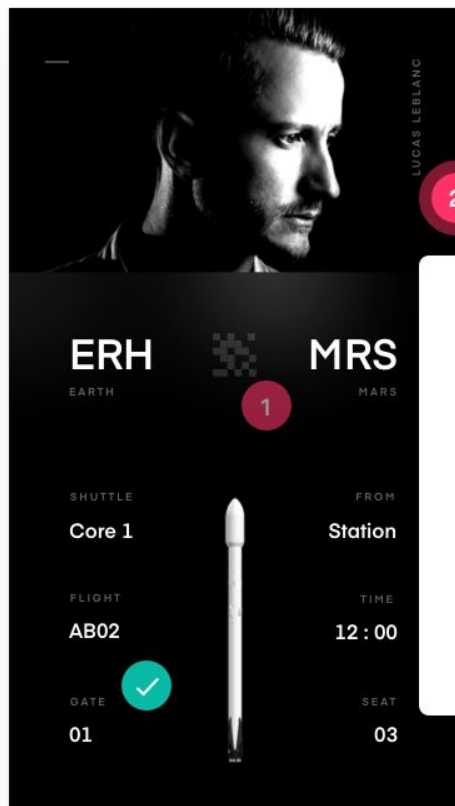
Button

Subhead 1

Ut rutrum quam eget enim ultricies sodales. Quisque efficitur erat a volutpat tristique. Fusce condimentum purus in urna scelerisque volutpat. Duis porttitor massa nec enim.

Button

Ut rutrum quam eget enim ultricies sodales. Quisque efficitur erat a volutpat tristique. Fusce condimentum purus in urna scelerisque volutpat. Duis porttitor massa nec enim malesuada, vitae aliquam neque vulputate. Morbi dictum ex porta, condimentum diam et, placerat magna.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus tincidunt augue suscipit orci hendrerit malesuada id et tellus.
Ut rutrum quam eget enim ultricies sodales. Quisque efficitur erat a volutpat tristique.

LUCAS LEBLANC

2

ERH
EARTH

MRS
MARS

1

SHUTTLE
Core 1

FROM
Station

FLIGHT
AB02

TIME
12 : 00

GATE
01

SEAT
03

Resolve

Isabelle Grant 20m ago

Is this meant to be a link?

2 likes

Cool, I'll work on it @isabellegrant

Post · Cancel

1 person will be notified. Add more

goodapi.co

SHARE
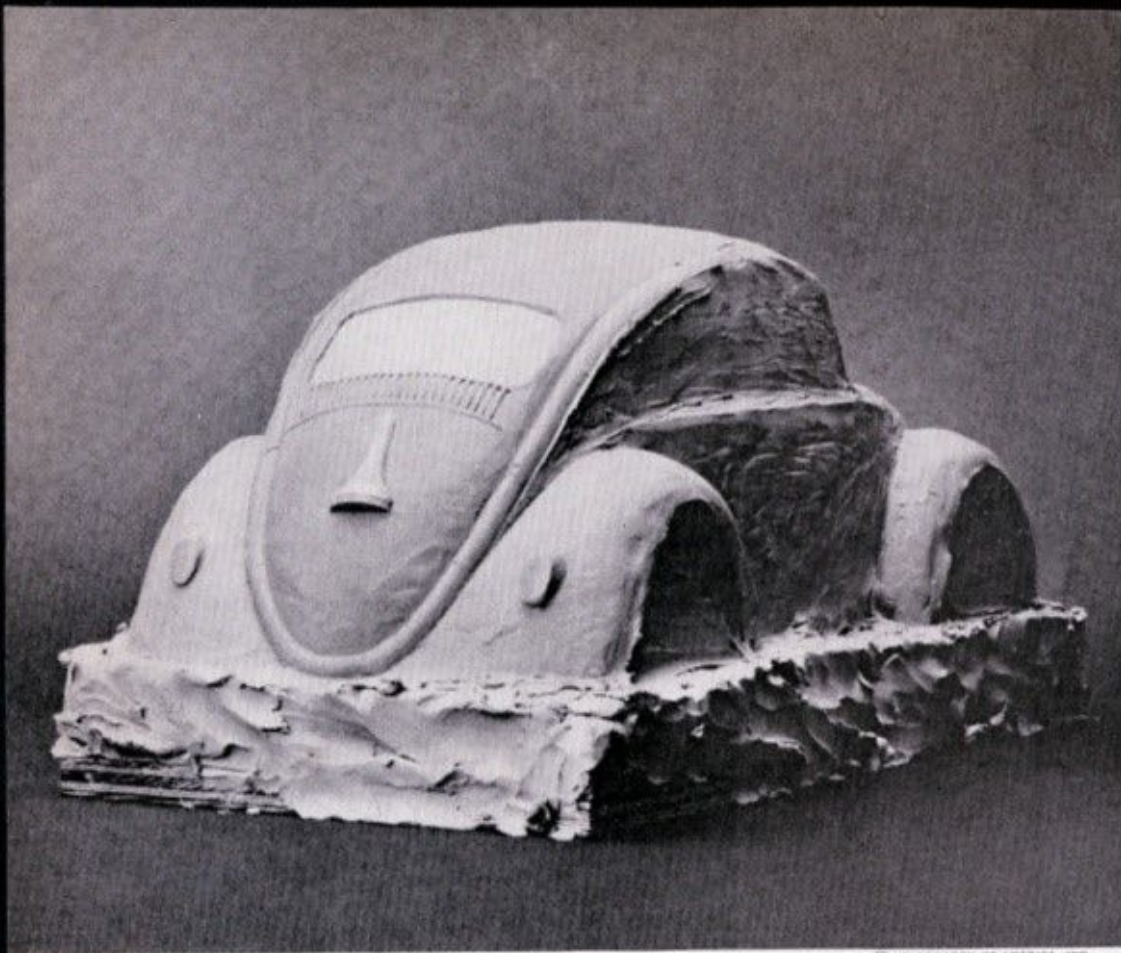
© VOLKSWAGEN OF AMERICA, INC.

We don't have to start from scratch each year.

API Design Contract based prototyping

Stoplight

Products   Resources   Company   Pricing   Request Demo   Sign In →

OpenAPI Powered Mock Servers

Build APIs and accelerate API development with instant mock servers

✓ Instant Mocking   ✓ Virtualization   ✓ Request Validation   ✓ Customizable

Agile API Lifecycle

Feedback

1 — DESIGN
2 — DEVELOP & TEST
3 — DEPLOY
4 — PUBLISH
5 — OPERATE
6 — CONSUME
7 — ANALYZE
8 — UPDATE

Prototype
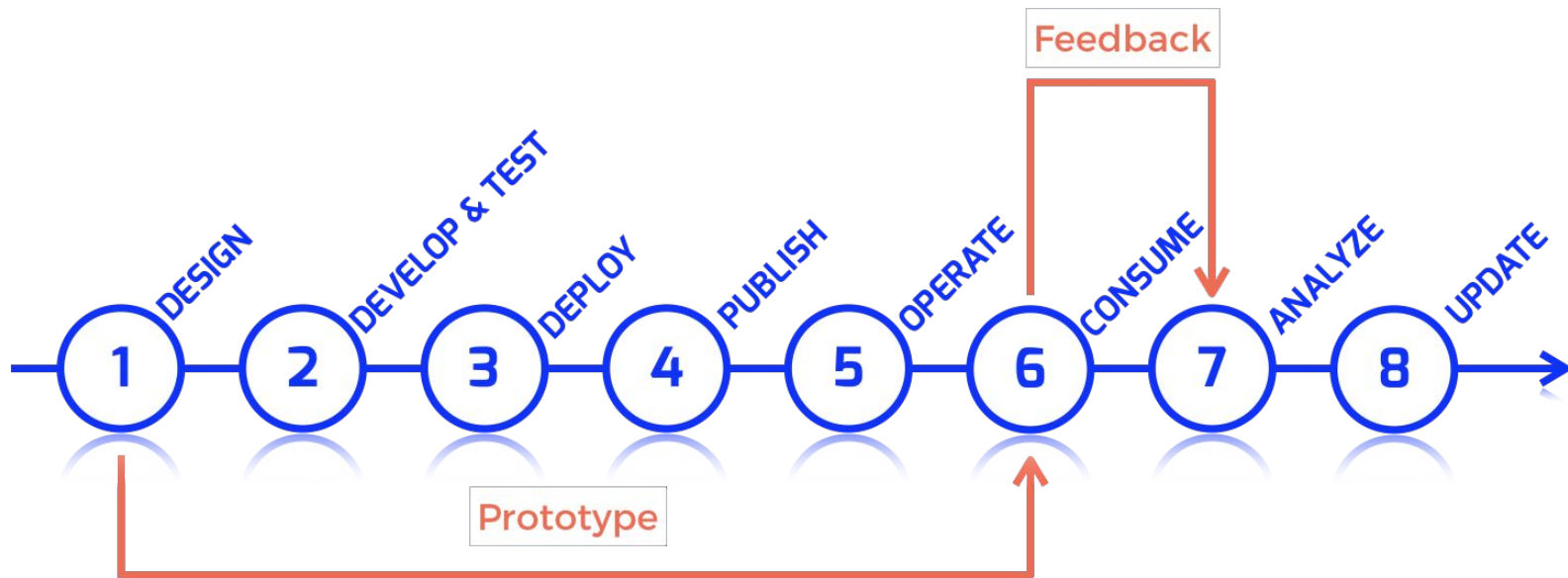
Evolving the API Product (design)

The first rule of Fight Club

Never break clients

Any modification to an existing API MUST avoid breaking changes and MUST maintain backward compatibility.

Rules for extending

1. You MUST NOT take anything **away**
2. You MUST NOT change **processing rules**
3. You MUST NOT make optional things **required**
4. Anything you add MUST be **optional**

Identifier Stability

URI is an identifier

Roy T. Fielding
@fielding

Follow

The reason to make a real REST API is to get evolvability … a "v1" is a middle finger to your API customers, indicating RPC/HTTP (not REST)

3:33 PM - 8 Sep 2013

277 Retweets 232 Likes

16   277   232
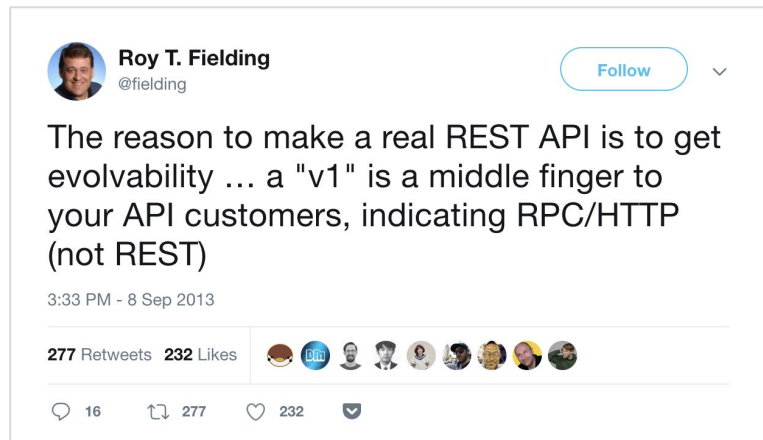
Backward incompatible changes

New Resource Variant

T0
Parameters `first` and `last` optional

T+1
Parameter `first` required

/greeting?first=John&last=Appleseed
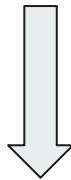
New Resource Variant

/named-greeting?first=John&last=Appleseed

Introduction to API Design

API Design

The Email Story

RE: RE: RE: RE: RE: RE: The API We Want

Have you ever tried to discuss what an API should look like over a plain email?
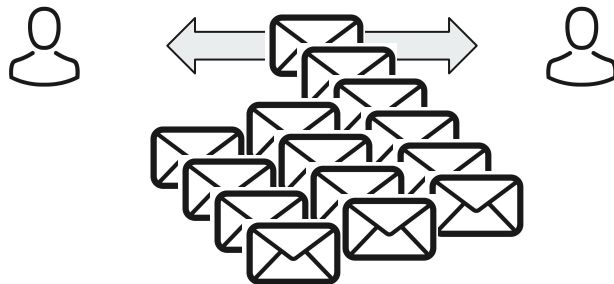
API Design

Good API Design

API design is API architectural style decision together with a set of additional design constraints applied on a specific domain, business, domain, organizational and technical requirements.

API Design

Good API Design is Verified

- Prototype
- Share
- Verify
    - Stakeholders
    - Users
    - API Developers

API Description

API Description

API Design is formalized using one of the API Description formats

- API Blueprint
- RAML
- Swagger
- OpenAPI Specification 2.0 (fka Swagger)
- OpenAPI Specification 3.0
- AsyncAPI Specification
- GraphQL Schema
- Avro Schema
- Protobuf

API Description

OpenAPI Specification 2.0 (formerly Swagger)

```yaml
swagger: '2.0'
info:
  description: This is a simple API
  version: 1.0.0
  title: Simple Inventory API
  # put the contact info for your development or API team
  contact:
    email: you@your-company.com

  license:
    name: Apache 2.0
    url: http://www.apache.org/licenses/LICENSE-2.0.html

# tags are used for organizing operations
tags:
- name: admins
  description: Secured Admin-only calls
- name: developers
  description: Operations available to regular developers

paths:
  /inventory:
    get:
      tags:
      - developers
      summary: searches inventory
      operationId: searchInventory
      description: |
        By passing in the appropriate options, you can search for
        available inventory in the system
      produces:
      - application/json
```

API Description

API Description must be:

- Human & machine readable
- Accessible
- Understood by Stakeholders, Developers, API Consumers, Tech writers, DevOPS, Support team
- Versioned in VCS
- Treated as Product Requirements

1 DESIGN

- Collect business requirements
- Identify Affordances & Resources
- Reconcile Vocabularies
- Pick architectural style
- Author API description
- Verify API description
- Check design consistency
- Approve API description - **contract**

2 DEVELOP & TEST

- Implement the API surface
- Test the API locally
- Verify the **contract** locally

3 DEPLOY

- Setup CI/CD pipeline
- Test the API in CI/CD
- Verify the **contract** in CI/CD
- Deploy the API
  - On-prem (Internal Cloud)
  - Public Cloud

4 PUBLISH

- Expose the API internally or externally based on the **contract**
- Security, throttling, consumption setup
- Operational & Business analytics setup

5 OPERATE

- Operational analytics
- Monitor security, throttling, consumption, availability, performance
- Scale
- Support users
- Resolve operational issues

6 CONSUME

- Publish API on Dev portal
- API documentation based on the **contract**
- Obtain API credentials (keys, tokens)
- Experiment with API
- Use the API
- User support

7 ANALYZE

- Business analytics
- Monitor per account usage, **contract** compliance
- Monetization

8 UPDATE

- Review the design
- Collect new business requirements
- Update the API description
- Approve the updated API description - **new contract**

1 — 2 — 3 — 4 — 5 — 6 — 7 — 8 →

goodapi.co

Contract-driven API Lifecycle

API Lifecycle Must be Contract-driven
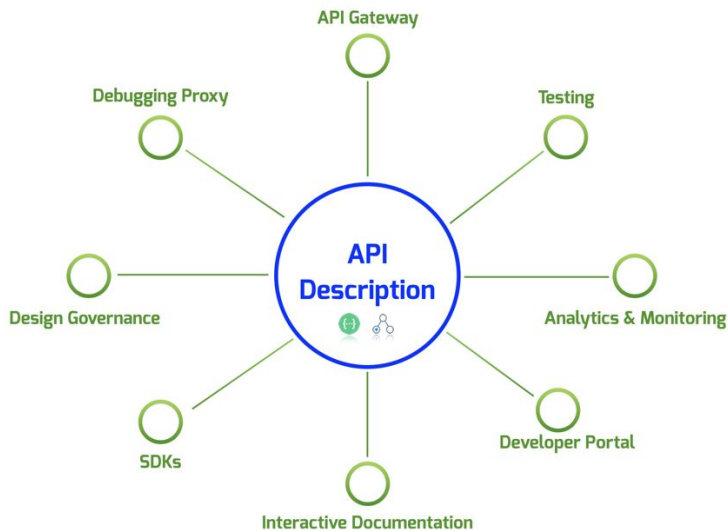
Keeping Things in Sync is Hard

INTERFACE **ACTUALLY** EXPOSED

are they the same?

INTERFACE **DOCUMENTED**

are they the same?

INTERFACED **USED**

are they the same?

Developer Portal

API Documentation

SDKs

Mock Service

Debugging Proxy

API User

API Client

API Catalogue

API Guidelines

VCS

Security Checks

CI / CD

Security Checks

API Design Governance Tests

API Description

API Implementation Souce Code

Runtime Platform

API Gateway

API Contract Tests

API Monitoring

API Analytics

API Monetization

Auth Provider

embeds

calls · uses · reads · registers at

calls

calls

calls

generates · generates · generates · generates

sync

triggers

triggers

triggers

verifies

drives

is listed

verifies

sync

verifies

configures

drives

verifies

configures · configures · configures

triggers

monitors

analyzes

reports to

authenticates

provides credentials

provides billing

calls

GOOD API

Independent API Consulting
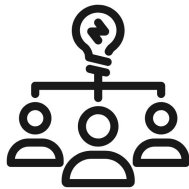
Approved API design becomes the contract

API-first

The Three Pillars

Three Pillars of Digital Transformation API-first

Business

Organization

Technology

Business

APIs are the products that you create, manage, sell, measure, recombine, and retire.
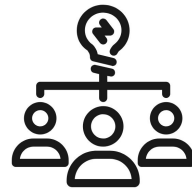
API First means that if a product in your organization does not have an API, it essentially does not exist.

APIs are the connective fabric.

Organization

No matter how good your service or product is, if it is delivered through a bad API, this substantially diminishes the quality of the product.

Teams embrace the fact that APIs is their main deliverables.

Technology

Everything that gets created or consumed in the organization is based on APIs.

The quality of APIs greatly matter for how easy it is to create and consume company products.

APIs change must not create expensive and time-consuming ripple effects through a chain of API dependencies.

APIs are no longer just about the technology

___

APIs are an exercise of cross-organizational collaboration

———

GOOD
API

Independent API Consulting