I'm Frank Kilcommins

› API Technical Evangelist @ SmartBear

› Developer and Architect passionate about APIs & Developer Experience

› Governance Board member on OpenAPI Initiative (OAI)

› Core contributor on Workflows Specification

› Connect:

/* insert embarrassing photo here */

@fkilcommins

@frank-kilcommins

Talk Track

› APIs – We Can't Escape Them

› Why APIs Fail

› What Makes a "Good API"

› Developer Experience

› API Specifications

› Leveraging BDD for API delivery

› Looking Ahead – Workflows for API

› Takeaways

› Q&A

What is an API?

APIs are not solely technical interfaces; they expose **digital products ready to be embedded** to **deliver** and **export capabilities** enabling new customer experiences

Allow **People** Interact — All **People** Inclusive

Are there more appropriate words for the *API* acronym?

Allow **Participants** — Access Products

Innovate   Interoperably

The API Scene

API Landscape – Trends / Recent Inflections

› **The great digital shift**

10 years of digitalization in under **one** year

34%

2020
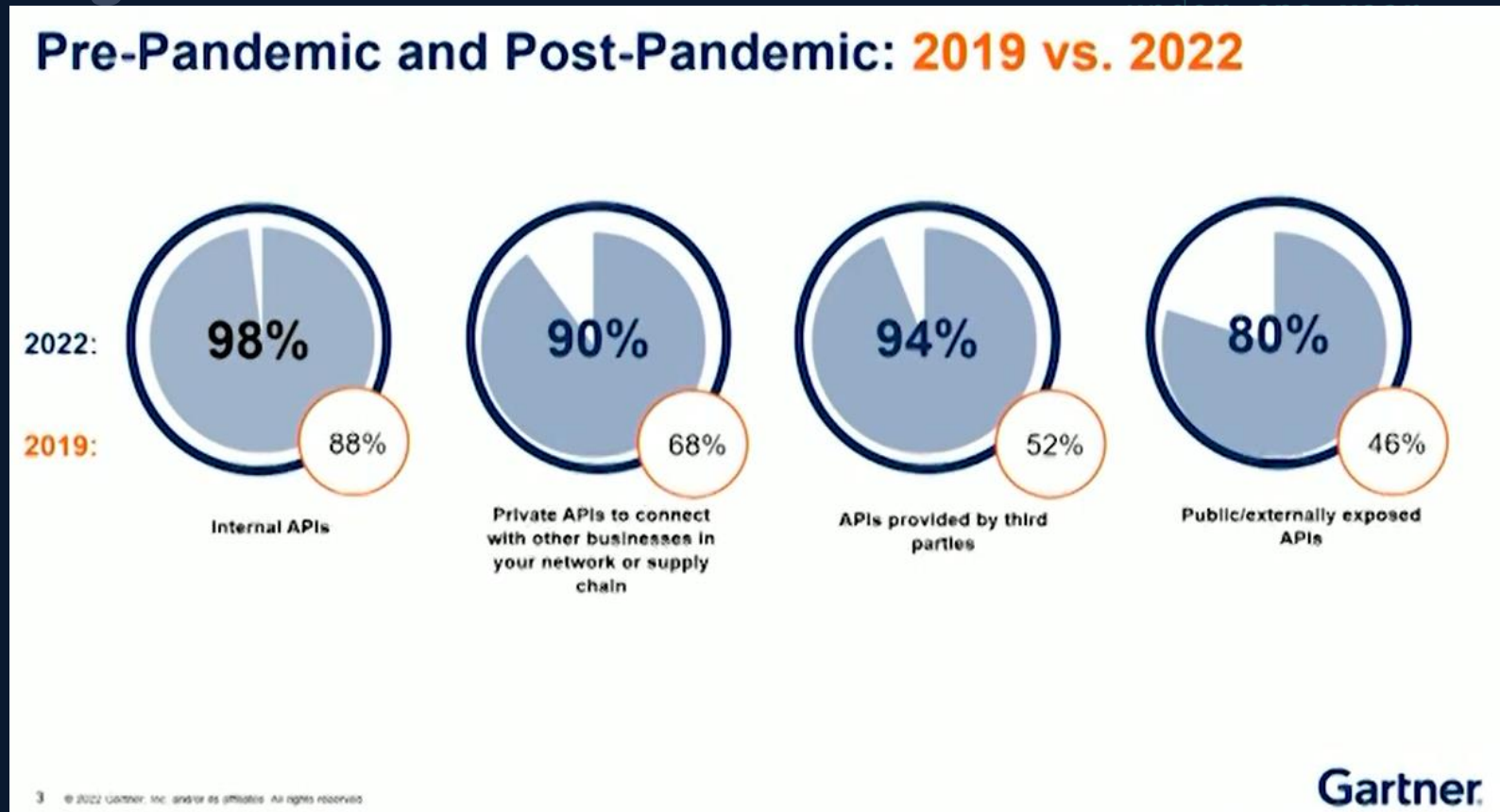
16%

2019

% of business channels replaced by digital
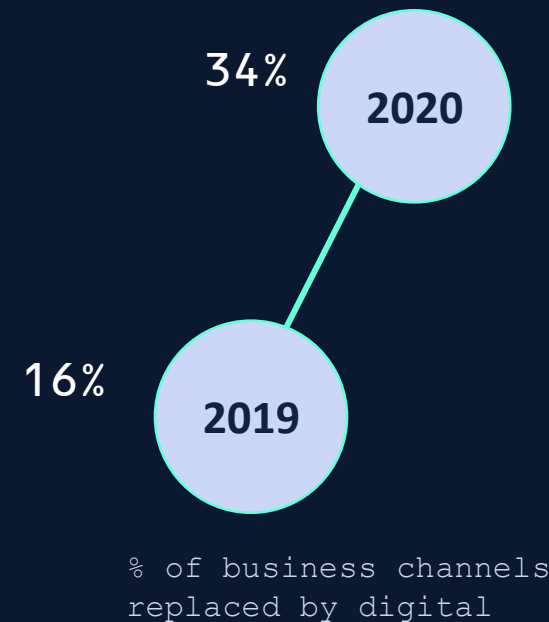
Source: Mckinsey & Company

API Landscape – Trends / Recent Inflections

› **The great digital shift**

10 years of digitalization in

Pre-Pandemic and Post-Pandemic: 2019 vs. 2022

2022:	98%	90%	94%	80%
2019:	88%	68%	52%	46%
	Internal APIs	Private APIs to connect with other businesses in your network or supply chain	APIs provided by third parties	Public/externally exposed APIs

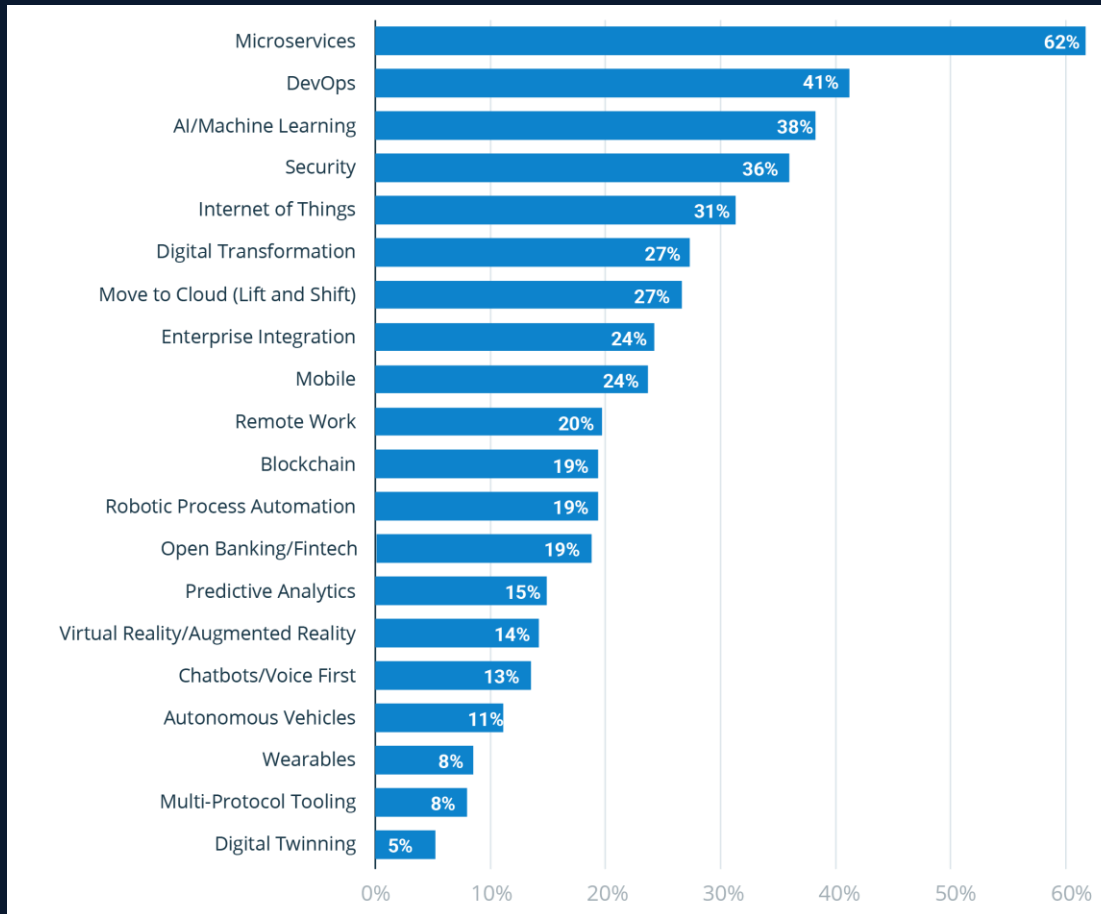# API Landscape – Trends / Recent Inflections

› The great digital shift

› **Growing complexity**

80% of organization estimate they have up
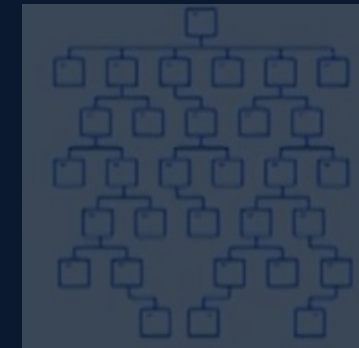to **1000** applications in their portfolio
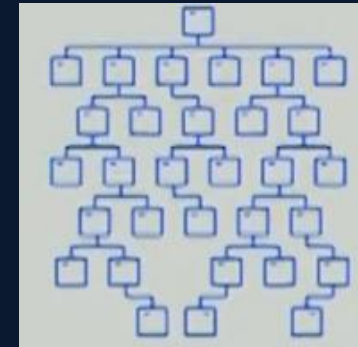
Source: IDC Worldwide Application Services 2021

API Landscape – Trends / Recent Inflections

SmartBear – State of Software Quality | API 2023

80% of organization estimate they have up to **1000** applications in their portfolio

Source: IDC Worldwide Application Services 2021
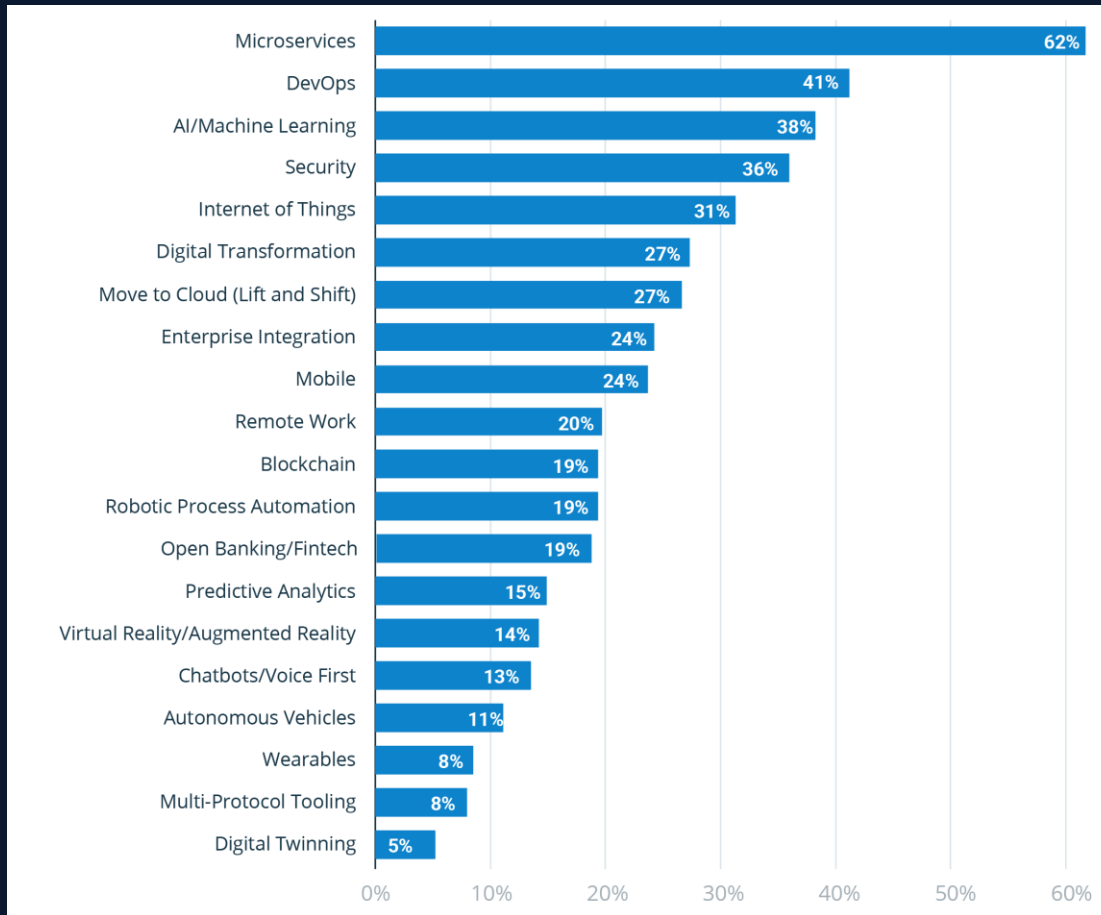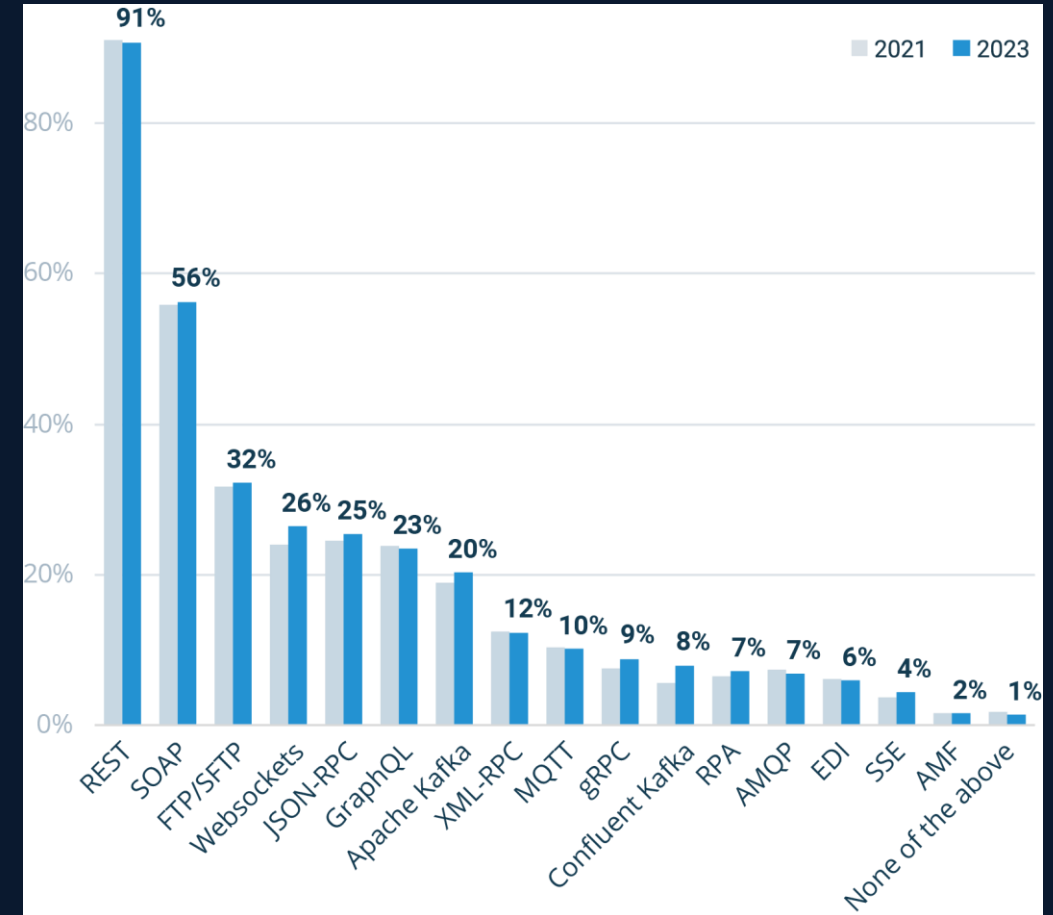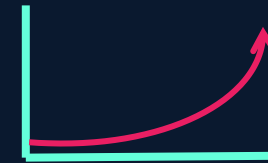
API Landscape – Trends / Recent Inflections

Microservices	62%
DevOps	41%
AI/Machine Learning	38%
Security	36%
Internet of Things	31%
Digital Transformation	27%
Move to Cloud (Lift and Shift)	27%
Enterprise Integration	24%
Mobile	24%
Remote Work	20%
Blockchain	19%
Robotic Process Automation	19%
Open Banking/Fintech	19%
Predictive Analytics	15%
Virtual Reality/Augmented Reality	14%
Chatbots/Voice First	13%
Autonomous Vehicles	11%
Wearables	8%
Multi-Protocol Tooling	8%
Digital Twinning	5%

2021 / 2023

REST 91%, SOAP 56%, FTP/SFTP 32%, Websockets 26%, JSON-RPC 25%, GraphQL 23%, Apache Kafka 20%, XML-RPC 12%, MQTT 10%, gRPC 9%, Confluent Kafka 8%, RPA 7%, AMQP 7%, EDI 6%, SSE 4%, AMF 2%, None of the above 1%

SmartBear – State of Software Quality | API 2023

SmartBear – State of Software Quality | API 2023

API Landscape – Trends / Recent Inflections

› The great digital shift

› Growing complexity

› **Increasing threats**

Attacks on web applications increased
to make up **39%** of all breaches in 2021

Verizon 2021 Data Breach Investigations Report

API Landscape – Trends / Recent Inflections

› The great digital shift

› Growing complexity

Attacks on web applications increase
to make up **39%** of all breaches in 2021

› **Increasing threats**

"API attacks up ???% in first half of 2021"

SALT

API Landscape – Trends / Recent Inflections

› The great digital shift

› Growing complexity

› **Increasing threats**

Attacks on web applications increase
to make up **39%** of all breaches in 2021
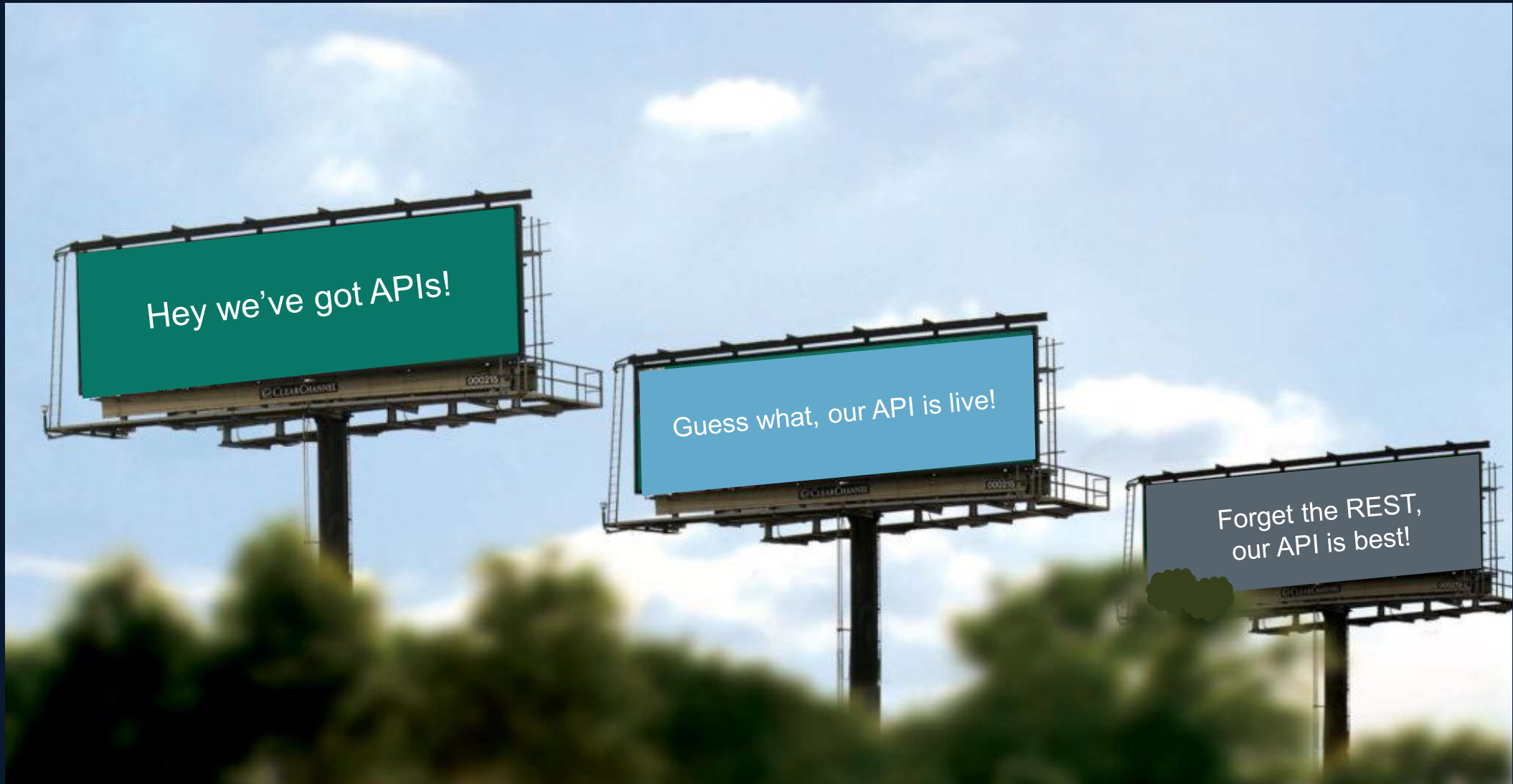
"API attacks up **348%** in first half of 2021"

SALT

Why APIs Fail

Just Having APIs is Not Enough!

Why APIs Fail

› **Organizational challenges**
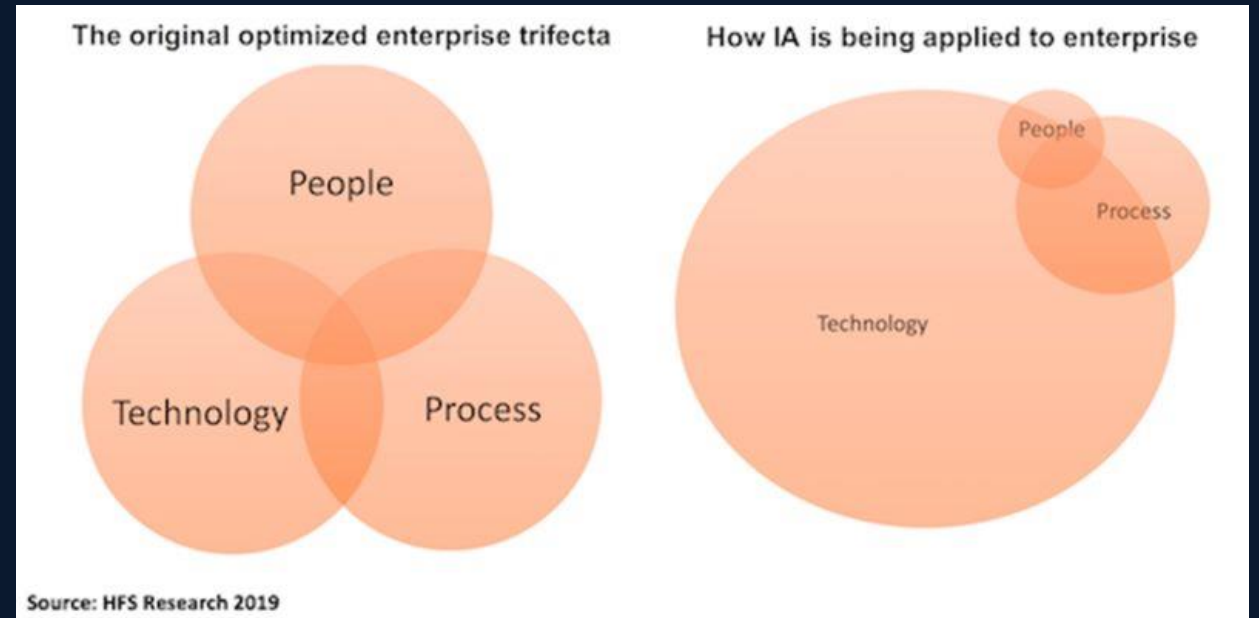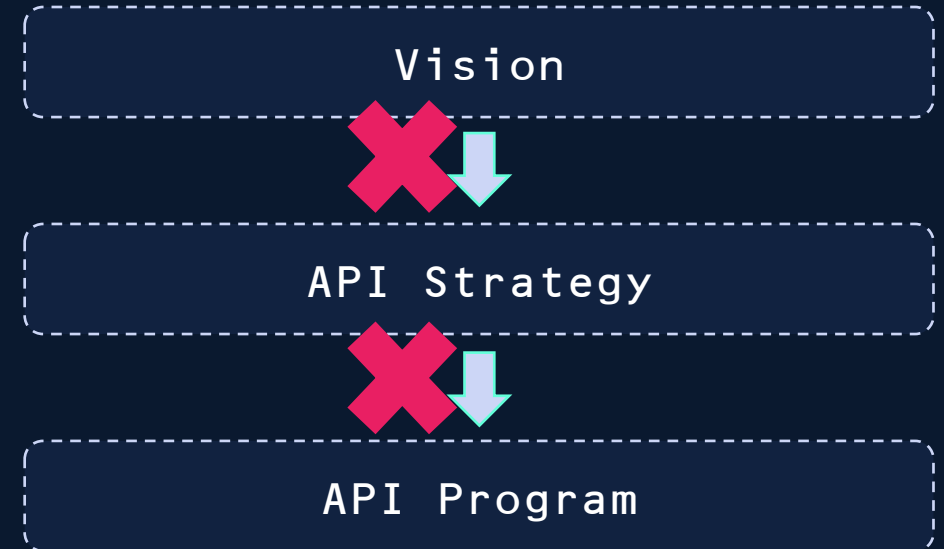

Technology focus dominates rather than people and process.
(This graphic reflects Integrated Automation but is also true for APIs)

Why APIs Fail

› Organizational challenges
› **Execution gaps**

What Failure Looks Like

› **APIs are hard to find**

What Failure Looks Like

› APIs are hard to find

› **Messed up security**

What Failure Looks Like

› APIs are hard to find
› Messed up security
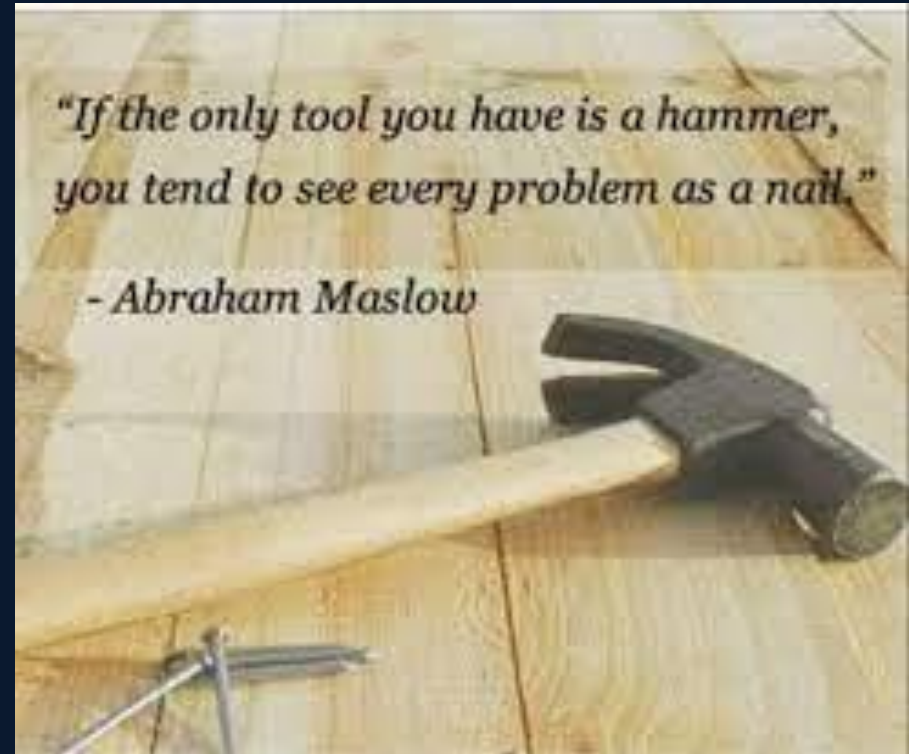› **Poor / No documentation**

What Failure Looks Like

› APIs are hard to find

› Messed up security

› Poor / No documentation

› **Poor error feedback**

What Failure Looks Like

› APIs are hard to find

› Messed up security

› Poor / No documentation

› Poor error feedback

› **Wrong API style selection**

"If the only tool you have is a hammer, you tend to see every problem as a nail."

- Abraham Maslow

What Failure Looks Like

› APIs are hard to find

› Messed up security

› Poor / No documentation

› Poor error feedback

› Wrong API style selection

› ...

› **Major version proliferation**

› **No support process**

What Makes a Good API?

Traits of Good APIs

› Designed collaboratively from the *outside-in*

› Easy to discover

› Easy to learn

› Easy to use

› Safe to use

› Easy to extend

› Secure

› Appropriate taxonomy

› Clear documentation

› Clear communication

› Long living

"Good APIs = Useful, Findable and Useable"
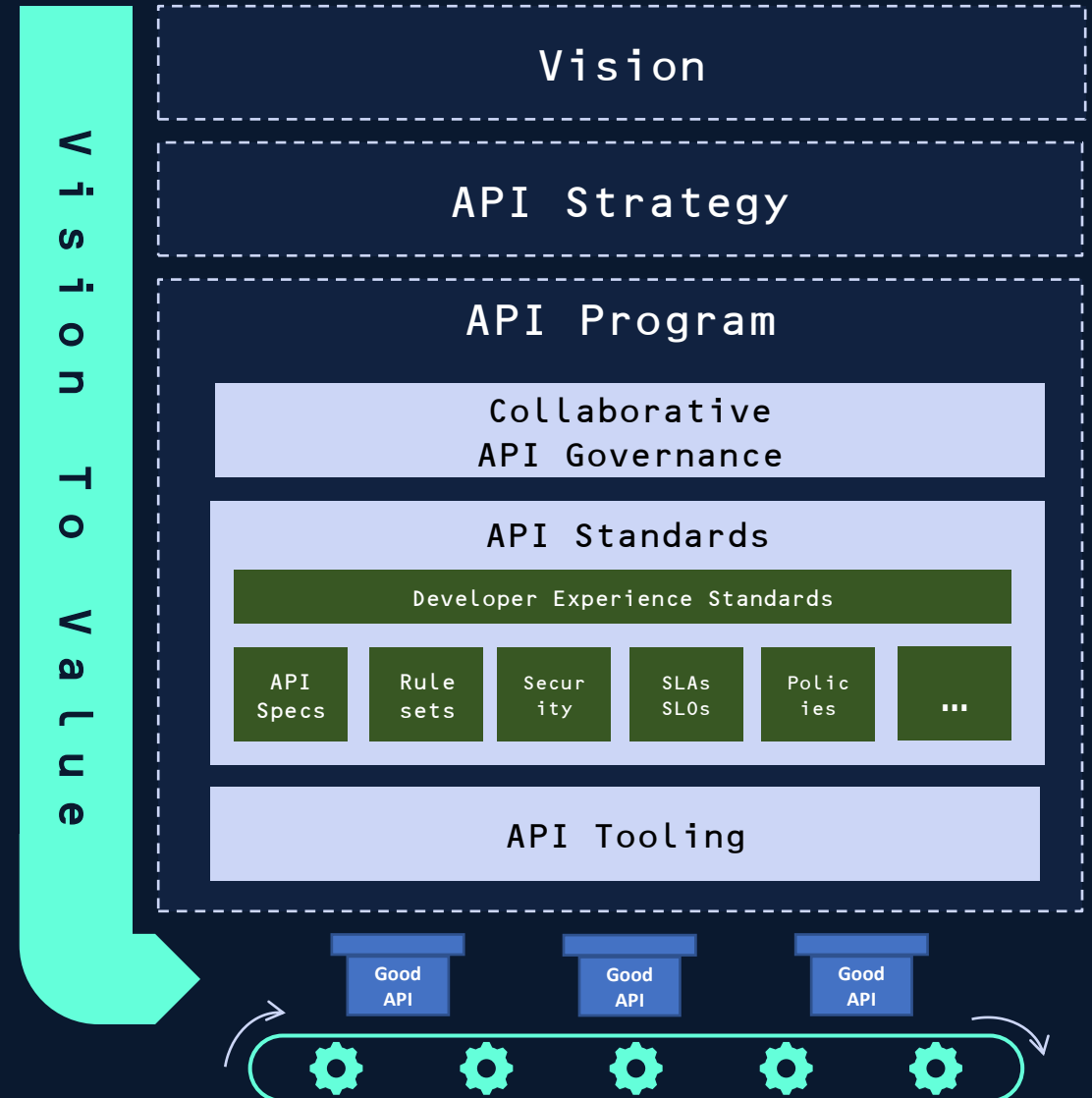Dr. Erik Wilde

Trait of a Successful API

› Longevity!

Keep the API Total Cost of Ownership perspective

Figuring out what to do   Doing it   Maintaining what we've done! (patching or evolving)

Scaling Good APIs

› Bridge the gaps

› Equip teams to delivery consistency

› Define success – appropriate KPIs

› Visibility over API portfolio

API Developer Experience (DX)

"We waste **11%–20%** of our time in front of our computers on systems that do not work or that are so difficult to understand that we cannot perform the task we want to."

University of Copenhagen & Roskilde University: Morten Hertzum et al, **Frustration: Still a Common User Experience,** *ACM Transactions on Computer-Human Interaction* **(2023)**

Photo by Aron Visuals on Unsplash

Even though customers buy this...   ...they *really* want this.
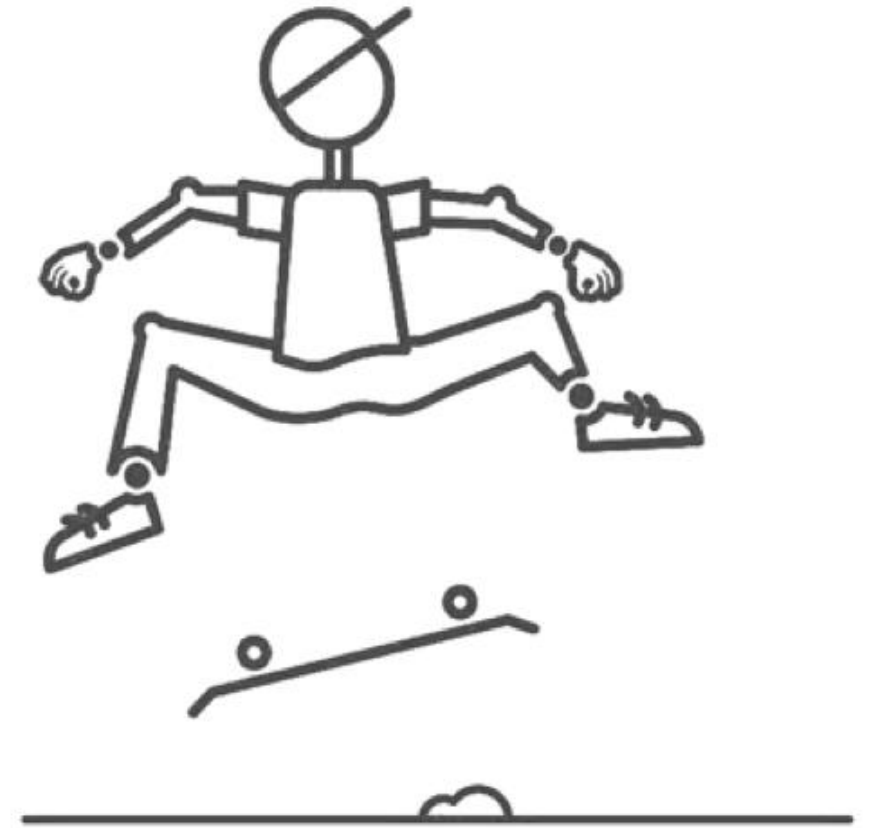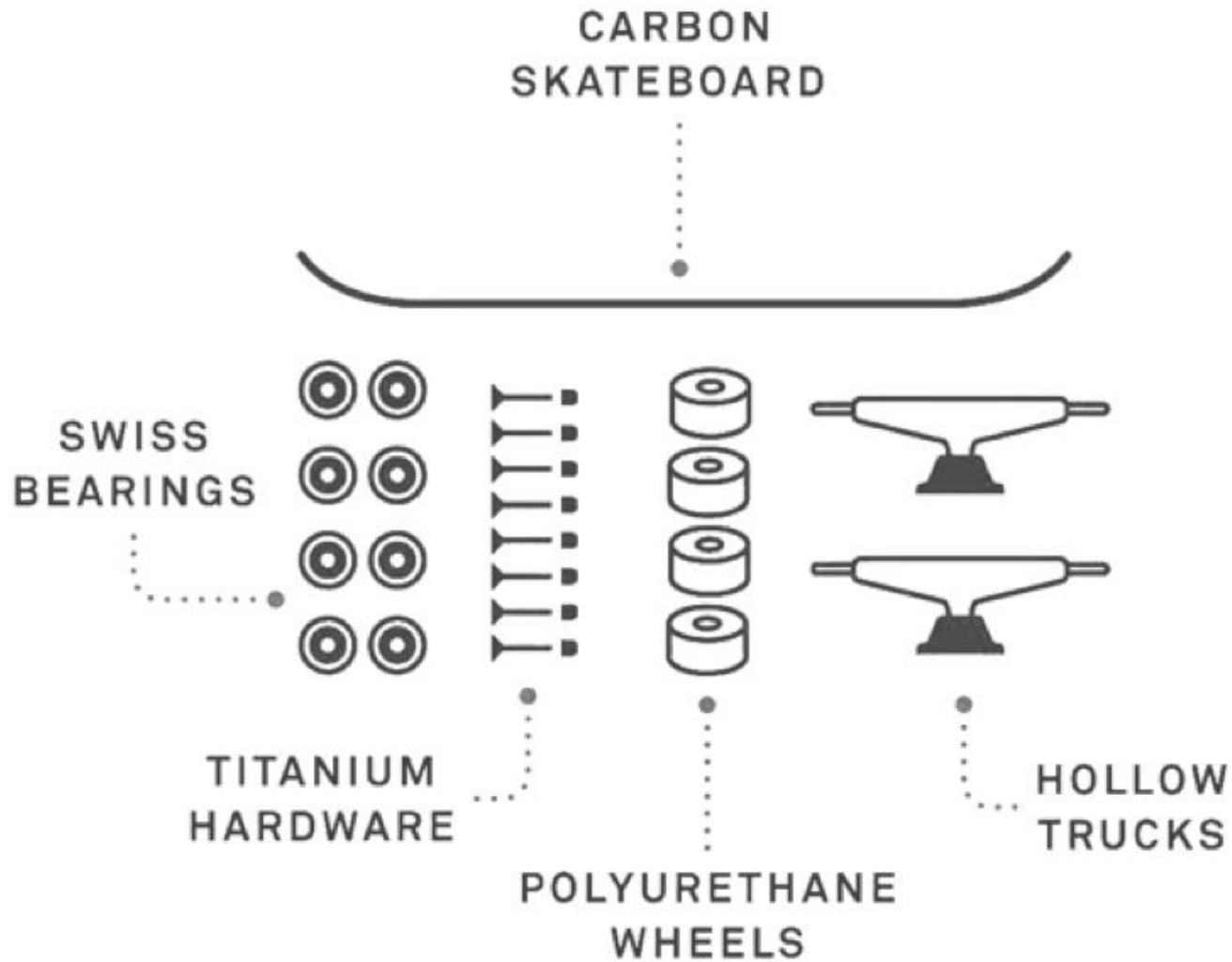
The designers at intercom (intercom.com) use this illustration to show what is, and isn't, important to customers.

Recipe for Good DX and Effective API Teams

› API-First as a guiding principle

› Standardisation & Governance
  › Style Guides
  › Patterns, Tooling, Reviews
  › Collaborative support

› Metrics for DX

› Focus on usability (empathy for consumer)
  › On-boarding
  › Use specifications
  › Provide examples, use-cases, recipes for success
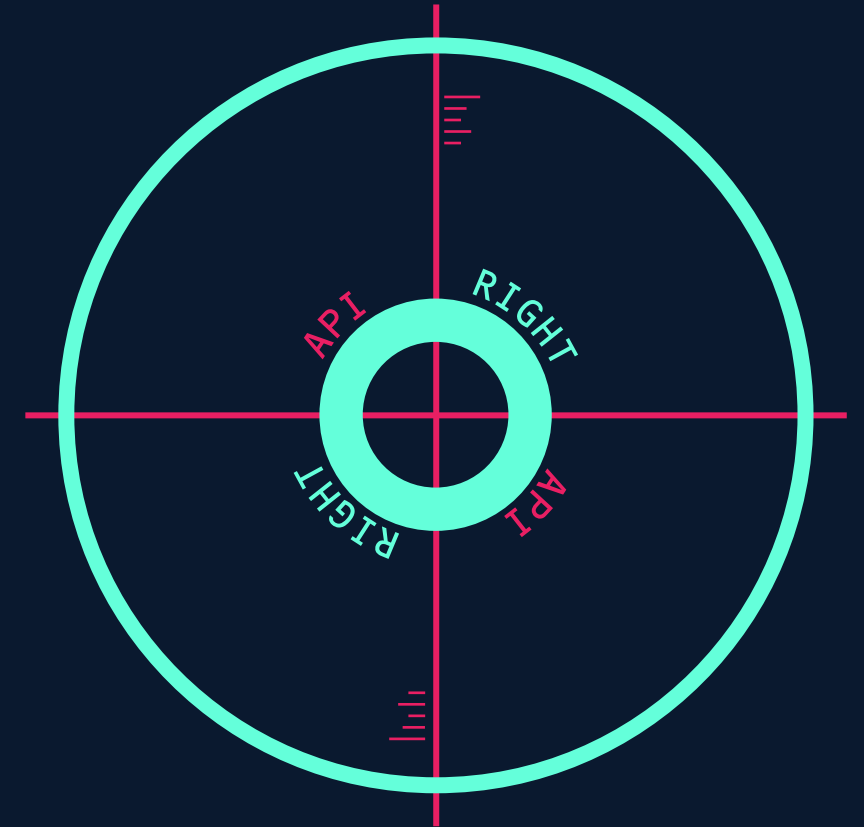
API Governance distilled:

1. Are we building the right API?

2. Are we building the API right?

API Specifications

"Good specifications will always improve programmer productivity far better than any programming tool or technique."
- Milt Bryce

Popular API related Specifications

Popular API related Specifications
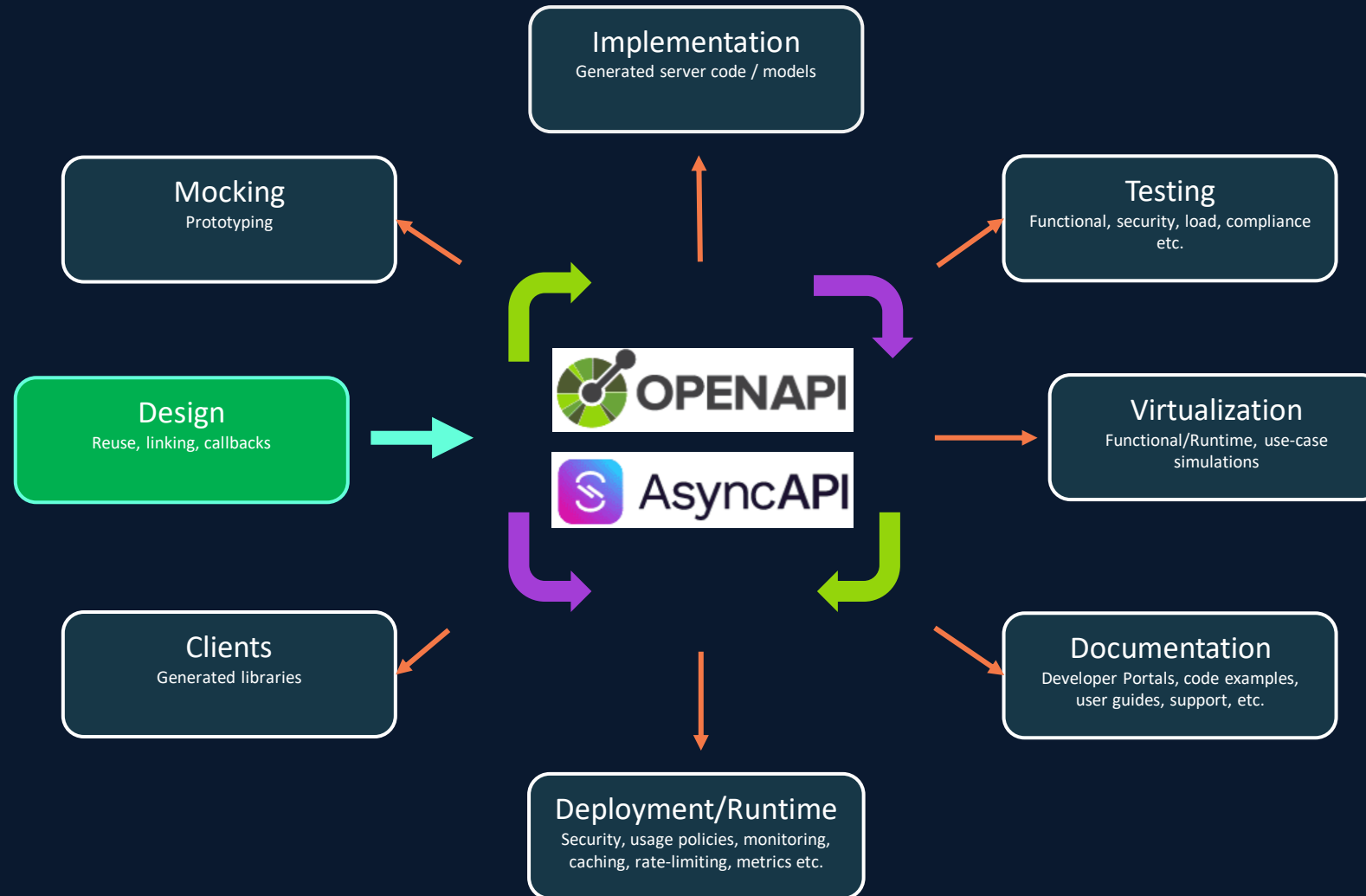
WSDL | WADL | RAML | API Blueprint

Let Specifications Drive the API Lifecycle

BDD for API Delivery

BDD for API Delivery
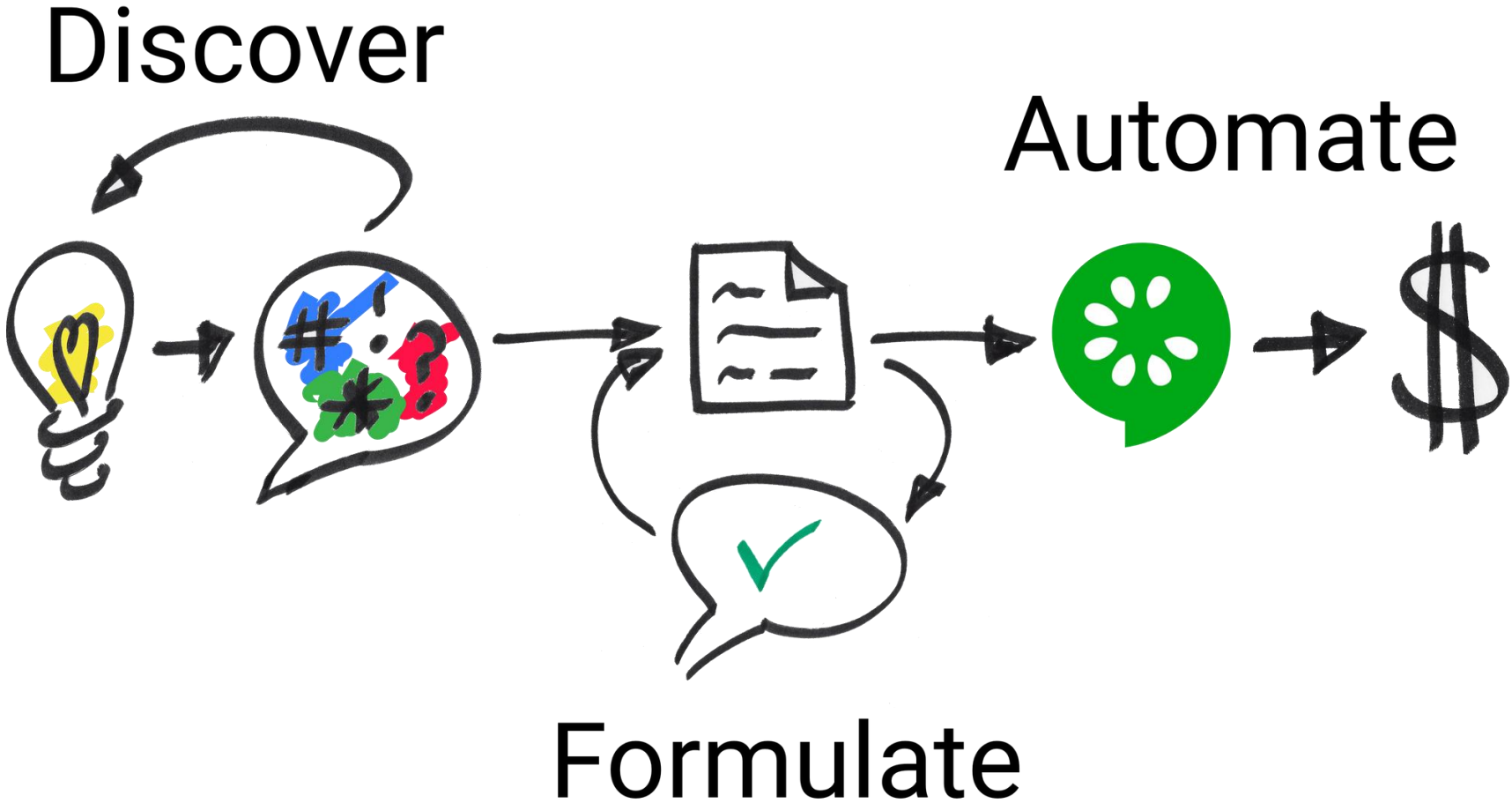
BDD for API Delivery

Discover
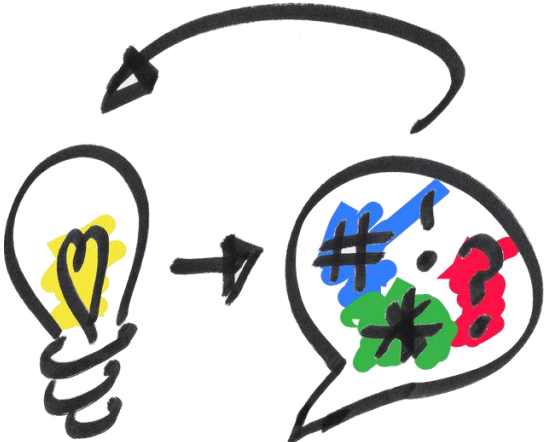
Automate

OPENAPI

Formulate

BDD for API Delivery
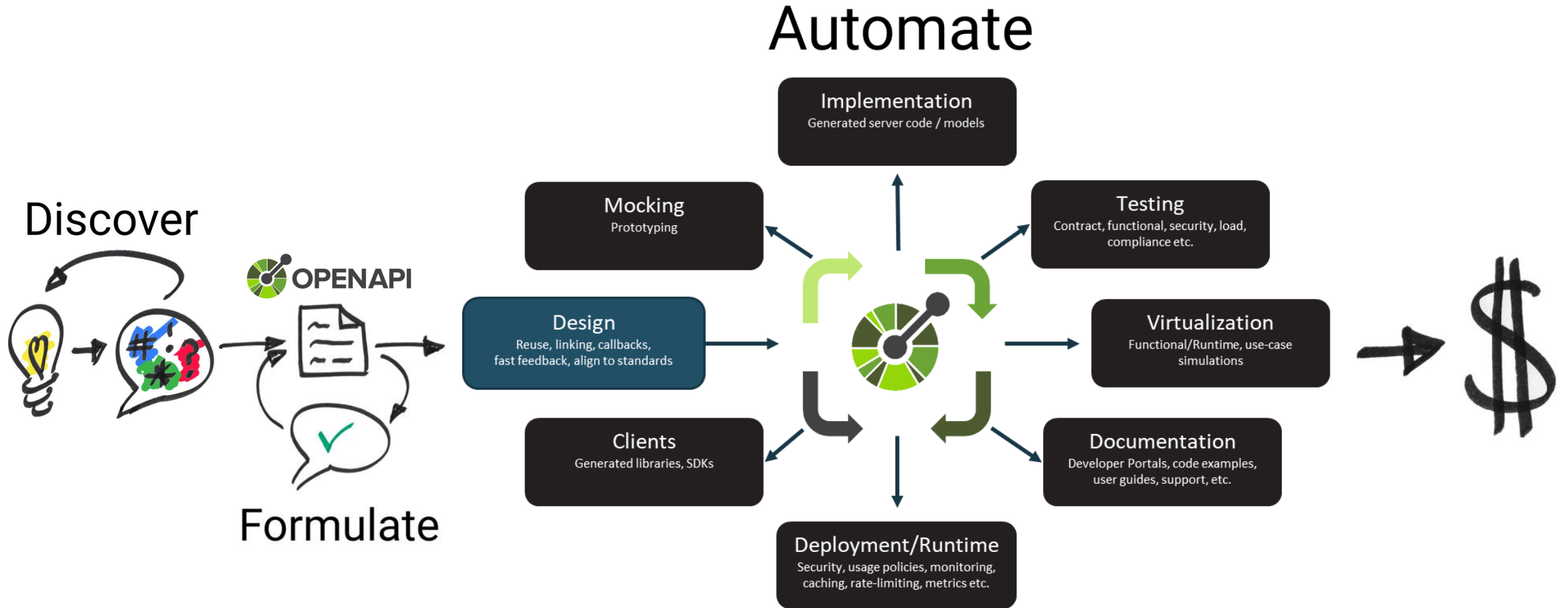
Example Mapping for Deliberate Discovery

› Effectively explore and refine detailed acceptance criteria so that:
  › **Agreed** the **scope**
  › **Identified** as much **uncertainty** as possible
  › **Agreed** on the **readiness** for design/development
  › Identified **concrete examples**

Example Mapping for Deliberate Discovery

› Effectively explore and refine detailed acceptance criteria so that:
   › **Agreed** the **scope**
   › **Identified** as much **uncertainty** as possible
   › **Agreed** on the **readiness** for design/development
   › Identified **concrete examples**

Unknown unknowns

Example Mapping for Deliberate Discovery

› Effectively explore and refine detailed acceptance criteria so that:
  › **Agreed** the **scope**
  › **Identified** as much **uncertainty** as possible
  › **Agreed** on the **readiness** for design/development
  › Identified **concrete examples**

Unknown unknowns → Known unknowns
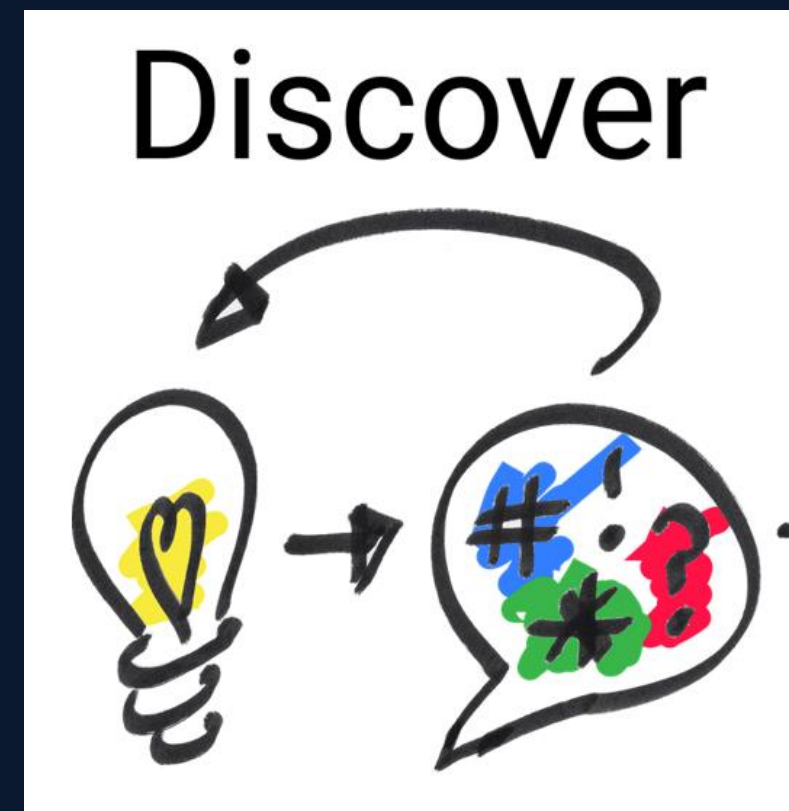
Example Mapping for Deliberate Discovery

› Effectively explore and refine detailed acceptance criteria so that:
  › **Agreed** the **scope**
  › **Identified** as much **uncertainty** as possible
  › **Agreed** on the **readiness** for design/development
  › Identified **concrete examples**

Unknown unknowns → Known unknowns → Known knowns

Who should participate?

Product Owner

QA

Developer

The traditional 3 Amigos

Who should participate?

Product Owner

QA

Developer

API Designer *

API Architect *
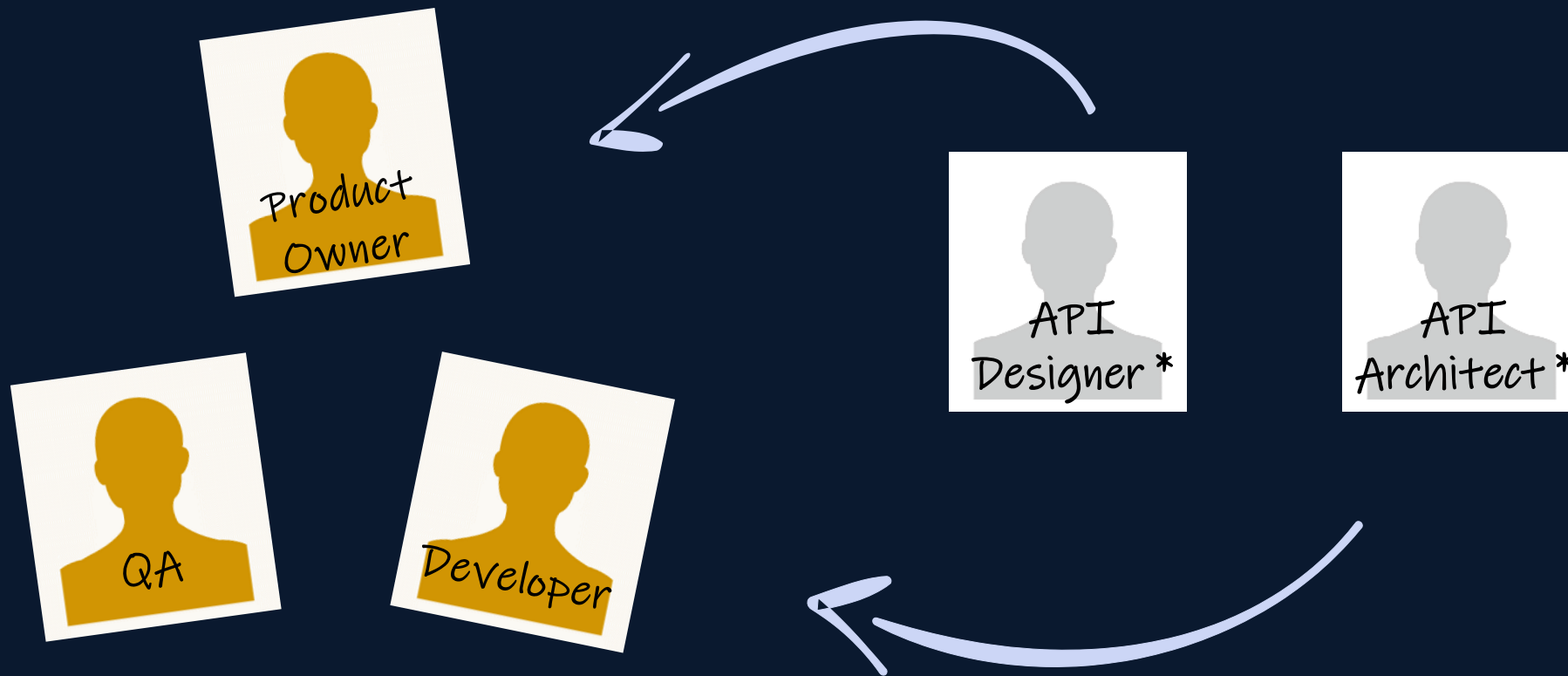
The traditional 3 Amigos

Enhanced with API specific SMEs
* dedicated or floating from a C4E

Creating a Pet Adoption Service

Pet Co

Problem:
We're overwhelmed by the number of pets being abandoned at our pet store.

Solution:
Following research and evaluation, we've decided to create the ability to search for and adopt pets that we catalog.

We'll offer the service via our website but also with a wider eco-system of shelters, charities, and pet orphanages.

Photo by Thomas Park on Unsplash

**Creating a
Pet Adoption Service**

Pet Co

Problem:
We're overwhelmed by the number of pets being abandoned at our pet store.

Solution:
Following research and evaluation, we've decided to create the ability to search for and adopt pets that we catalog.

We'll offer the service via our website but also with a wider eco-system of shelters, charities, and pet orphanages.

SEARCH FOR PETS

Photo by Thomas Park on Unsplash

SEARCH FOR PETS

CALLING THE SEARCH
PETS RESOURCE
RETURNS ALL
AVAILABLE PETS

PETS CAN BE FILTERED
BY STATUS, CATEGORY,
LOCATION, & BREED

SEARCH FOR PETS

CALLING THE SEARCH
PETS RESOURCE
RETURNS ALL
AVAILABLE PETS

PETS CAN BE FILTERED
BY STATUS, CATEGORY,
LOCATION, & BREED

JANE CALLS THE SEARCH
PETS RESOURCE
=> ALL DETAILS OF
AVAILABLE PETS ARE
RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

JANE CALLS THE SEARCH PETS RESOURCE => ALL SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS) *STANDARD*

JANE CALLS THE SEARCH PETS RESOURCE
=> ~~ALL~~ PAGED SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

JANE SEARCHES FOR ADOPTED STATUS PETS
=> PAGED SUMMARY OF ADOPTED PETS RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS) *STANDARD*

JANE CALLS THE SEARCH PETS RESOURCE
=> ~~ALL~~ PAGED SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

JANE SEARCHES FOR ADOPTED STATUS PETS
=> PAGED SUMMARY OF ADOPTED PETS RETURNED

JANE SEARCHES FOR PETS LOCATED IN GALWAY
=> PAGED SUMMARY OF GALWAY LOCATED PETS RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS) *STANDARD*

JANE CALLS THE SEARCH PETS RESOURCE => ~~ALL~~ PAGED SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

JANE SEARCHES FOR ADOPTED STATUS PETS => PAGED SUMMARY OF ADOPTED PETS RETURNED

JANE SEARCHES FOR PETS LOCATED IN GALWAY => PAGED SUMMARY OF GALWAY LOCATED PETS RETURNED

JANE SEARCHES CATS => PAGED SUMMARY OF CATS REGARDLESS OF LOCATION, STATUS, OR BREED RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS) *STANDARD*

WHEN CRITERIA DOES NOT MATCH A PET RETURN EMPTY RESPONSE (NOT ERROR) *STANDARD*

JANE CALLS THE SEARCH PETS RESOURCE => ~~ALL~~ PAGED SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

JANE SEARCHES FOR ADOPTED STATUS PETS => PAGED SUMMARY OF ADOPTED PETS RETURNED

JANE SEARCHES FOR PETS LOCATED IN GALWAY => PAGED SUMMARY OF GALWAY LOCATED PETS RETURNED

JANE SEARCHES CATS => PAGED SUMMARY OF CATS REGARDLESS OF LOCATION, STATUS, OR BREED RETURNED

JANE SEARCHES FOR BOXER BREED CATS

=> AN EMPTY RESPONSE IS RETURNED

SEARCH FOR PETS

CALLING THE SEARCH PETS RESOURCE RETURNS ALL AVAILABLE PETS

PETS CAN BE FILTERED BY STATUS, CATEGORY, LOCATION, & BREED

A PET SUMMARY SHOULD HAVE: NAME, ID, STATUS, CATEGORY, BREED, LOCATION

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS) *STANDARD*

WHEN CRITERIA DOES NOT MATCH A PET RETURN EMPTY RESPONSE (NOT ERROR) *STANDARD*
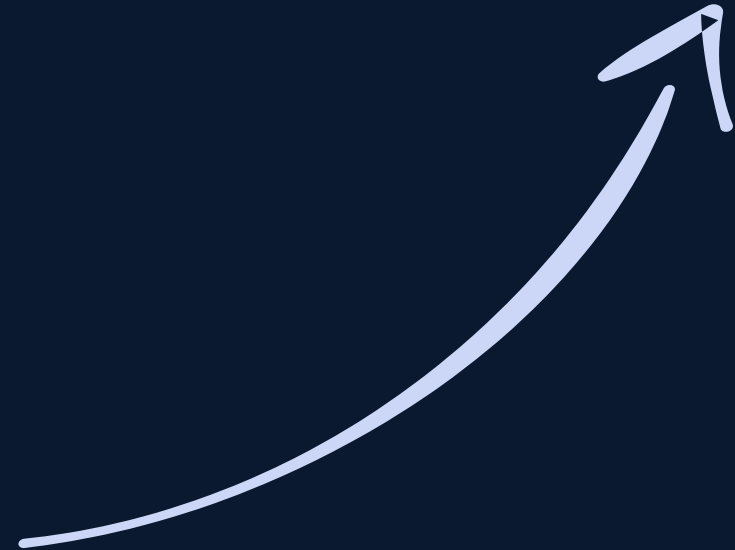
JANE CALLS THE SEARCH PETS RESOURCE => ALL PAGED SUMMARY DETAILS OF AVAILABLE PETS ARE RETURNED

JANE SEARCHES FOR ADOPTED STATUS PETS => PAGED SUMMARY OF ADOPTED PETS RETURNED

JANE SEARCHES FOR PETS LOCATED IN GALWAY => PAGED SUMMARY OF GALWAY LOCATED PETS RETURNED

JOE SEARCHES 1 PAGE OF PETS WITH 10 PETS * PAGE PARAMS VALIDATED => A PAGED RESULT CONTAINING 50 PETS WITH LINKS TO NEXT/SELF/PREVIOUS PAGES RETURNED

JANE SEARCHES CATS => PAGED SUMMARY OF CATS REGARDLESS OF LOCATION, STATUS, OR BREED RETURNED

JANE SEARCHES FOR BOXER BREED CATS

=> AN EMPTY RESPONSE IS RETURNED
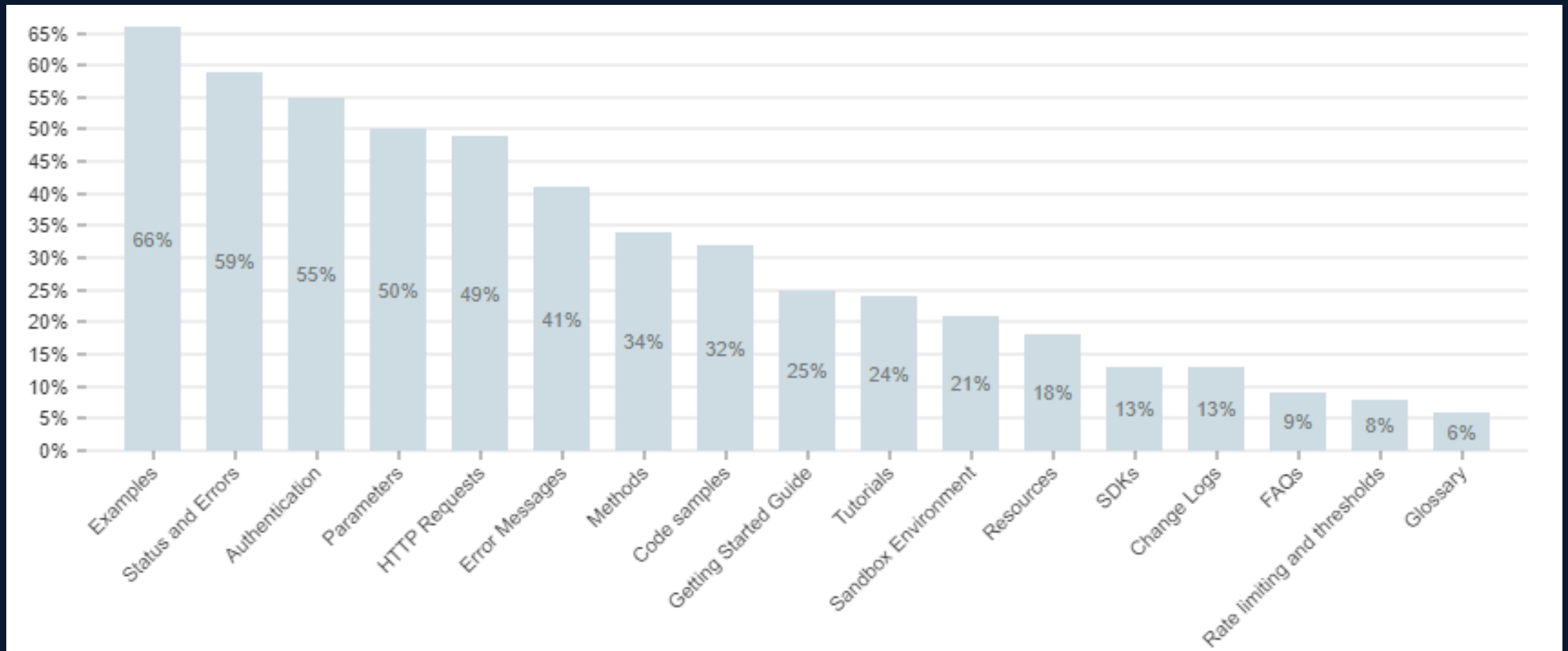
What does formulation look like?

What does formulation look like?

⚠️ Warning proprietary tool being used for demonstration

Most Important API Documentation Features?

› Examples continue to be the best learning resource for consumers

Most Important API Documentation Features?

› Examples continue to be the best learning resource for consumers
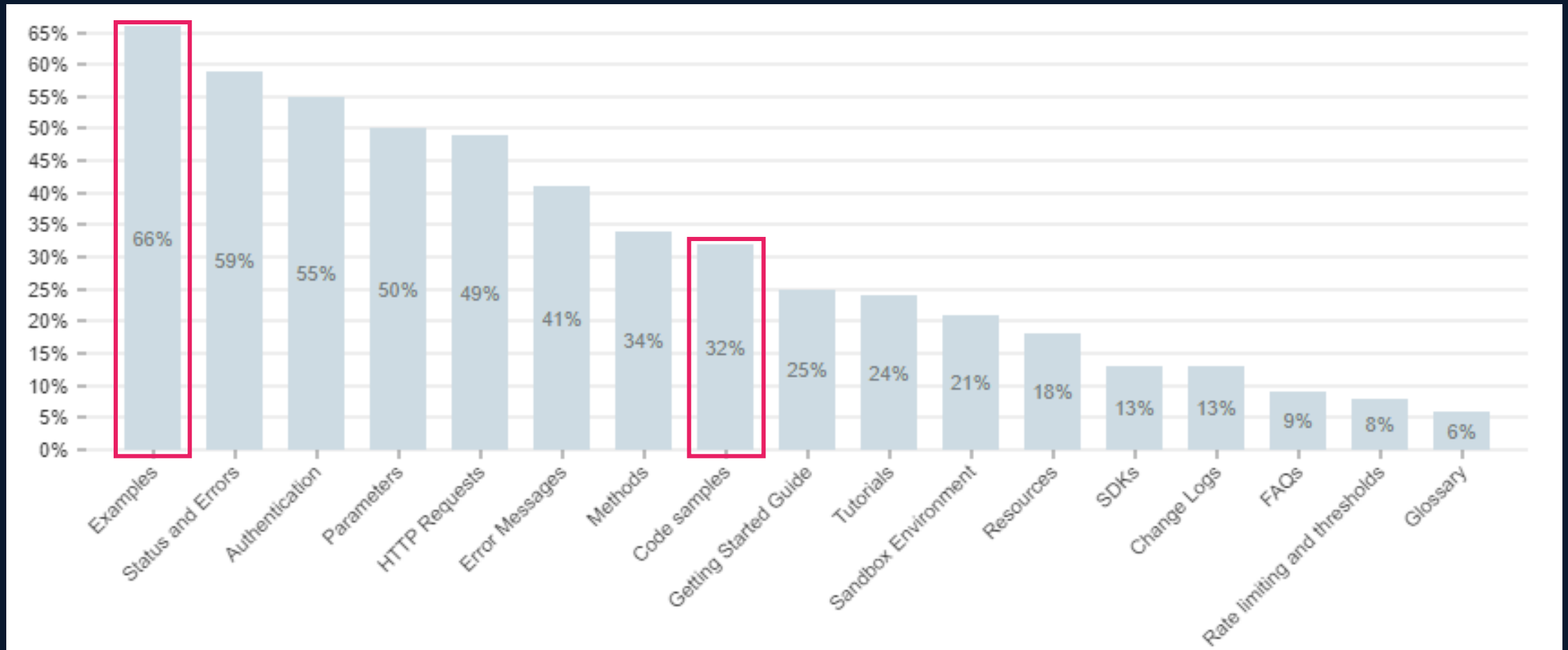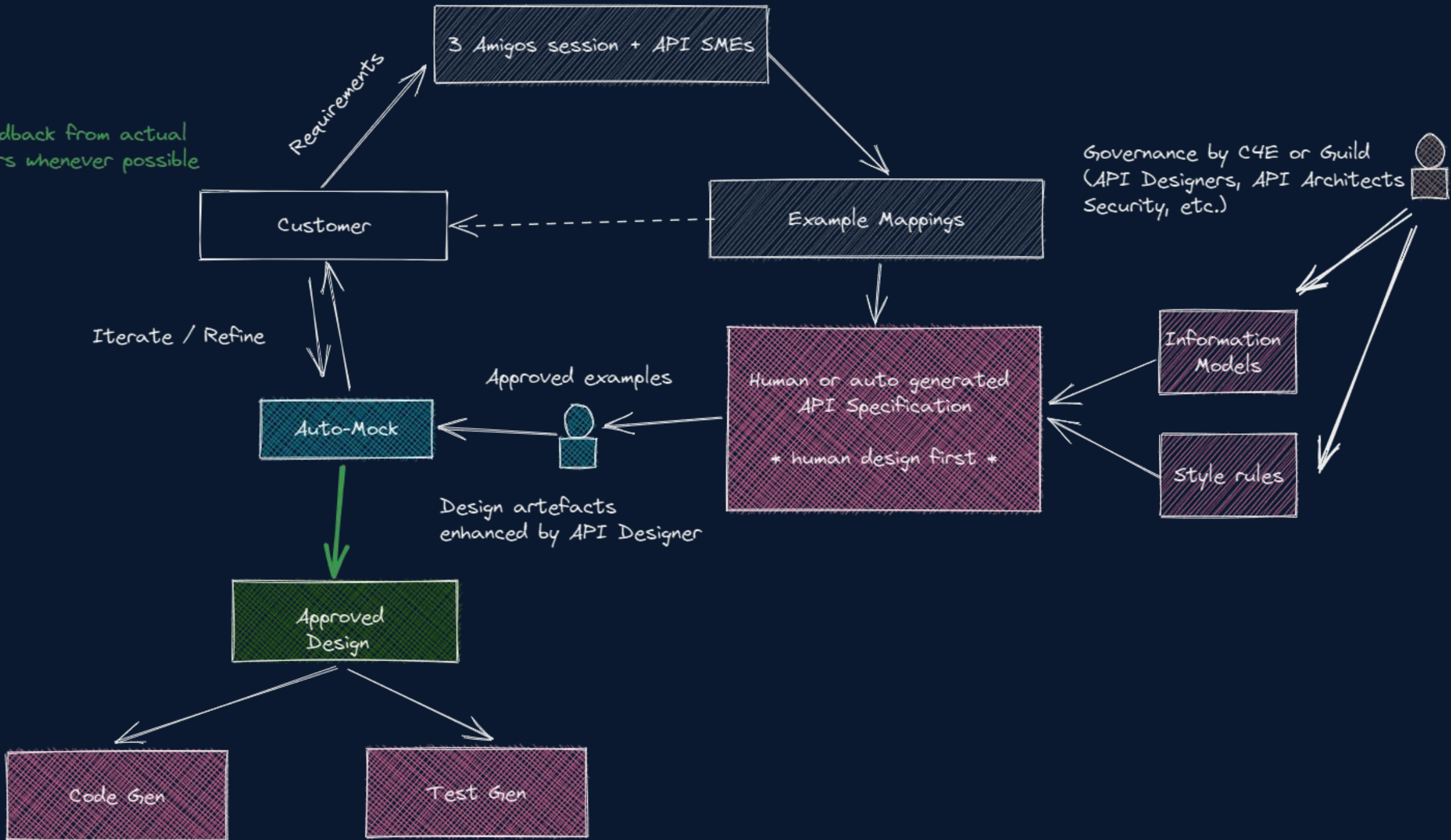
VIEW PET DETAILS

THE FULL PET DETAILS CAN BE RETURNED ONCE THE PETID IS KNOWN

JANE HAS THE PETID AND USES IT TO GET THE PET DETAILS
=> FULL DETAILS ON THE PET RETURNED

JOE USES THE NAME OF A PET TO GET THE DETAILS

=> A BAD REQUEST RESPONSE IS RETURNED

JOE HAS A PETID BUT IT DOES NOT MATCH A RECORD IN OUR SYSTEM
=> A NOT FOUND RESPONSE IS RETURNED

REQUEST PARAMETERS MUST MATCH THE EXPECTED TYPE OR PATTERN   *STANDARD*

VALID REQUESTS ON SINGULAR RESOURCE NOT MATCHING A RECORD MUST RETURN 404   *STANDARD*

ADD A NEW PET

AUTHENTICATED USERS CAN ADD NEW PETS FOR ADOPTION

JANE HAS AN AUTH TOKEN AND SENDS A NEW PET RECORD
* REQUEST VALIDATED
=> A SUCCESSFUL RESPONSE IS RETURN INCLUDING THE PET RESOURCE

JOE TRIES TO CREATE A PET BUT FORGETS HIS TOKEN
=> A UNAUTHORIZED REPONSE RETURNED

A NEW PET MUST HAVE AT LEAST NAME, CATEGORY, BREED, AND LOCATION SET

JOE SENDS A NEW PET RECORD WITH JUST NAME AND CATEGORY
* REQUEST VALIDATION FAILS
=> A BAD REQUEST IS RETURNED

ALWAYS RETURN A HTTP 201 RESPONSE FOR CREATED RESOURCES
STANDARD

JOE SENDS A NEW PET RECORD THAT MEETS VALIDATION RULES
=> A HTTP 201 RESPONSE IS CREATED WHICH ALSO CONTAINS THE PET RECORD

ALWAYS RETURN A LOCATION HTTP RESPONSE HEADER FOR CREATED RESOURCES
STANDARD

JOE SENDS A NEW PET RECORD THAT MEETS VALIDATION RULES
=> THE RESPONS CONTAINS A LOCATION HEADER WITH THE LINK TO THE GET /PETS/{ID}

ERROR RESPONSES MUST CONFORM TO RFC 7807
STANDARD

ADOPT A PET

AVAILABLE PETS CAN BE ADOPTED BY REGISTERED USERS

JANE IS REGISTERED AND CHOOSED TO ADOPT AN AVAILABLE PET

=> THE ADOPTION IS REGISTERED, THE PET STATUS IS UPDATED, AND A CONFIRMATION IS RETURNED TO JANE

SALLY IS UNREGISTERED AND TRIES TO ADOPT A PET

=> AN ERROR MESSAGE IS RETURNED TO SALLY AS SHE MUST BE REGISTERED

SHOULD A 401 UNAUTHORISED OR 403 FORBIDDEN STATUS BE RETURNED FOR UNREGISTERED?

JOE TRIES TO ADOPT AN ADOPTED PET => AN ERROR IS RETURNED WITH INFO THAT THE PET IS UNAVAILABLE FOR ADOPTION

JOE TRIES TO ADOPT AN UNKNOWN PET => A 404 NOT FOUND ERROR IS RETURNED TO JOE

GET PET ADOPTIONS LIST

A REGISTERED USER CALLING THE SEARCH ADOPTIONS ALL RECORDED ADOPTIONS

ADOPTIONS CAN BE FILTERED BY STATUS, AND LOCATION

A ADOPTION CAN HAVE STATUS OF REQUESTED, PENDING, AVAILABLE, DENIED, APPROVED

UNKNOWN QUERY PARAMETERS SHOULD BE IGNORED

STANDARD

SEARCHING PETS MUST SUPPORT PAGINATION (PAGED RESULTS & PARAMETERS)

STANDARD

WHEN CRITERIA DOES NOT MATCH A PET RETURN EMPTY RESPONSE (NOT ERROR)

STANDARD

JANE IS REGISTERED AND CALLS THE SEARCH ADOPTIONS RESOURCE
=> PAGED SUMMARY OF ADOPTIONS RETURNED

JANE SEARCHES FOR ADOPTIONS WITH PENDING STATUS
=> PAGED SUMMARY OF PENDING ADOPTIONS RETURNED

JOE SEARCHES FOR ADOPTIONS BUT OMITS TOKEN
=> AN UNAUTHORIZED RESPONSE IS RETURNED

JOE SEARCHES FOR INACTIVE ADOPTIONS
=> AN ERROR IS RETURNED AS INVALID IS NOT AN ADOPTION STATUS

JOE SEARCHES FOR ADOPTIONS WITH A COLOUR=RED
=> A PAGED SUMMARY OF ADOPTIONS RETURNED

Approved Formulation onto Automation

⚠️ Warning proprietary tool being used for demonstration

CHANGING YOUR API IN

DESIGN   VS   PRODUCTION

Modeling Flows in APIs

Workflows Specification

› The Workflows Specification describes an approach to document use-case oriented workflows in programmatically readable format.

› A workflow is a series of API calls which together accomplish some business objective.

› The additional human readable nature improves the ability of API specifications to *tell the story* of the API(s) in a manner that can improve the consuming developer experience.
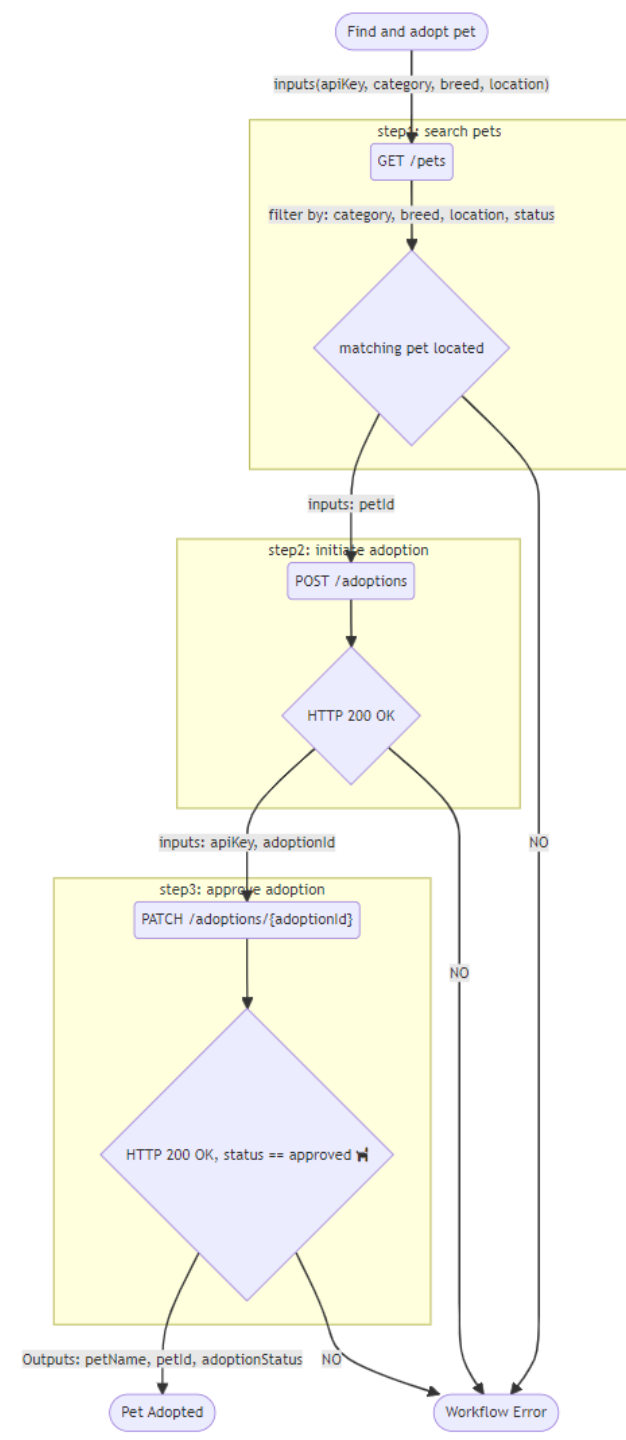
```yaml
workflowSpec: 1.0.0
info:
  title: A pet adoption workflow
  summary: This workflow showcases how to search for and adoption a pet from the PetCo `Pet Adoptions API`.
  description: |
      This workflow walks you through the steps of `searching` for, `selecting`, and `adopting` an available pet.
  version: 1.0.0
sources:
- name: adoptionsAPIDocument
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Pets-Adoptions-API/1.0.0/swagger.json
  type: openapi
workflows:
- workflowId: FindAndAdoptPet
  summary: Search for appropriate available pet and adopt it
  description: This workflow lays out the steps and API calls needed to search for an adopt a Pet
  inputs:
    type: object
    properties:
      apiKey:
        type: string
      category:
        type: string
      breed:
        type: string
      location:
        type: string
    required:
    - apiKey
  steps:
  - stepId: searchPets
    description: This step demonstrates the search pets flow for the first pet matching the criteria
    operationId: adoptionsAPIDocument.getPets
    parameters:
    - name: category
      in: query
      value: $inputs.category
    - name: breed
      in: query
      value: $inputs.breed
    - name: location
      in: query
      value: $inputs.location
    - name: page
      in: query
      value: 1
    - name: pageSize
      in: query
      value: 1
    - name: status
      in: query
      value: available
    successCriteria:
    - $statusCode == 200
    - $response.body.page.totalElements == 1
    outputs:
      petId: $response.body.data.pets[0].id
      petName: $response.body.data.pets[0].name
      petLocation: $response.body.data.pets[0].location
```

```yaml
workflowSpec: 1.0.0
info:
  title: A pet adoption workflow
  summary: This workflow showcases how to search for and adoption a pet from the PetCo `Pet Adoptions API`.
  description: |
        This workflow walks you through the steps of `searching` for, `selecting`, and `adopting` an available pet.
  version: 1.0.0
sources:
- name: adoptionsAPIDocument
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Pets-Adoptions-API/1.0.0/swagger.json
  type: openapi
workflows:
- workflowId: FindAndAdoptPet
  summary: Search for appropriate available pet and adopt it
  description: This workflow lays out the steps and API calls needed to search for an adopt a Pet
  inputs:
    type: object
    properties:
      apiKey:
        type: string
      category:
        type: string
      breed:
        type: string
      location:
        type: string
    required:
    - apiKey
  steps:
  - stepId: searchPets
    description: This step demonstrates the search pets flow for the first pet matching the criteria
    operationId: adoptionsAPIDocument.getPets
    parameters:
    - name: category
      in: query
      value: $inputs.category
    - name: breed
      in: query
      value: $inputs.breed
    - name: location
      in: query
      value: $inputs.location
    - name: page
      in: query
      value: 1
    - name: pageSize
      in: query
      value: 1
    - name: status
      in: query
      value: available
    successCriteria:
    - $statusCode == 200
    - $response.body.page.totalElements == 1
    outputs:
      petId: $response.body.data.pets[0].id
      petName: $response.body.data.pets[0].name
      petLocation: $response.body.data.pets[0].location
```

Takeaways

Summary

› The API landscape is complex

› Failure is common

› Focus on the **consumer**

› Keep **people** at the center of API delivery

› Human conversations & techniques like example mapping work

› **Collaboration** is key to delivering good Developer Experience

› Your API is a promise and let it tell its story

Thank You!
Questions?

› Connect:

@fkilcommins

@frank-kilcommins

frank.kilcommins@smartbear.com

Agile Alliance

PRESENTS

Agile2023

THE SCOTLAND EXPERIENCE

Join Agile Alliance today!

Become an Agile Alliance member and help support our non-profit mission, while gaining access to valuable benefits like online events, in-person conference discounts, and event session videos.