The Arazzo Specification:

Unlocking API Value for Humans and Machines

Frank Kilcommins / SmartBear

LangDev CON 2024

FS9U

ProxyHands

Seville 17-19 October, 2024

https://langdevcon.org

Arazzo

I'm Frank Kilcommins

- Principal API Technical Evangelist @ SmartBear
- Software Engineer and Architect (♥ APIs & Developer Experience)
- Governance Board member on OpenAPI Initiative (OAI)
- Contributor on the Arazzo Specification

- Connect:

/* insert embarrassing photo here */

@fkilcommins

@frank-kilcommins

frank.kilcommins@smartbear.com

SMARTBEAR

Arazzo

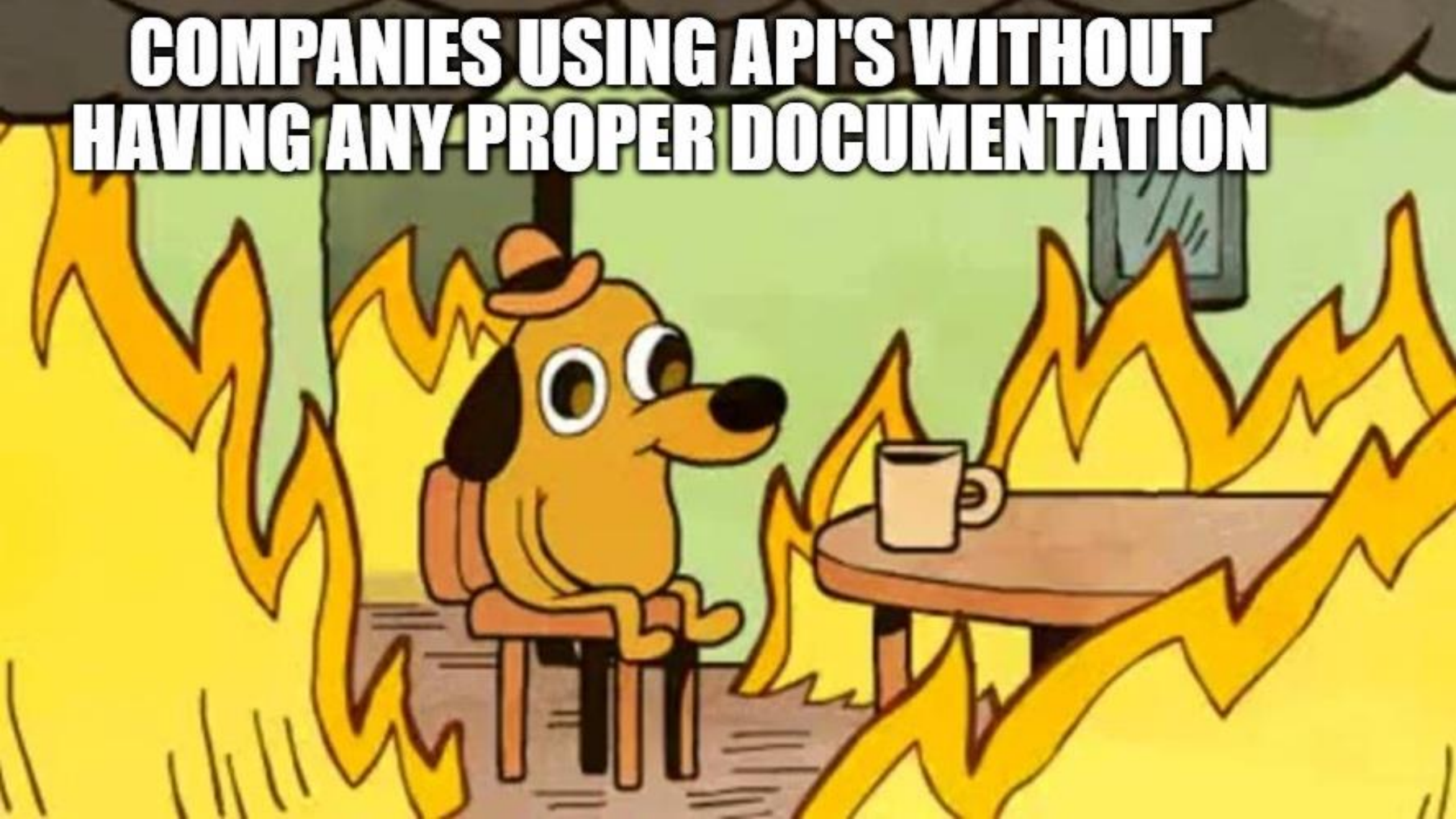
Agenda

- What is the Arazzo Specification
- Dominant use cases
- Arazzo Specification structure
- Example
- Q&A

FS9U

ProxyHands

SMARTBEAR


COMPANIES USING API'S WITHOUT
HAVING ANY PROPER DOCUMENTATION

Arazzo

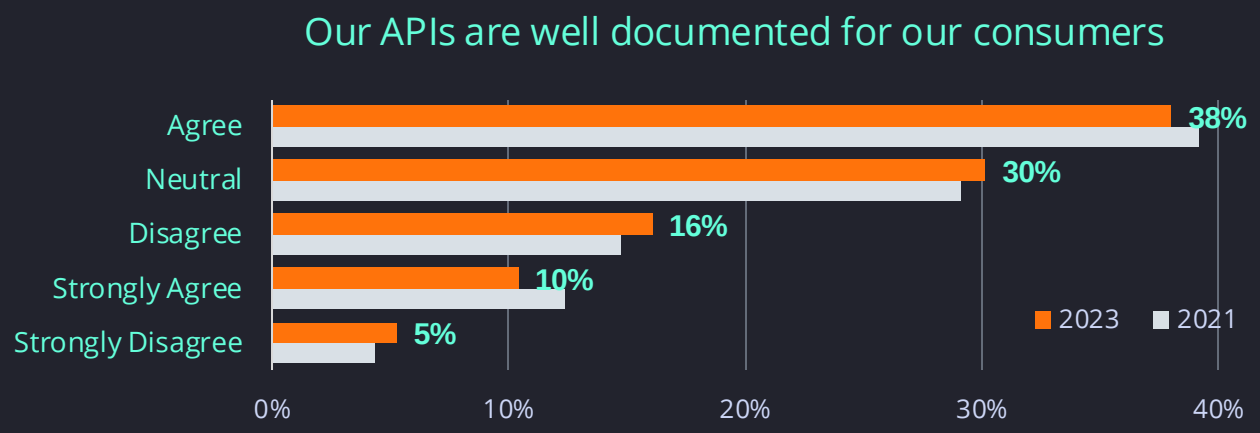
GIF by Butler University

OPENAPI

AsyncAPI

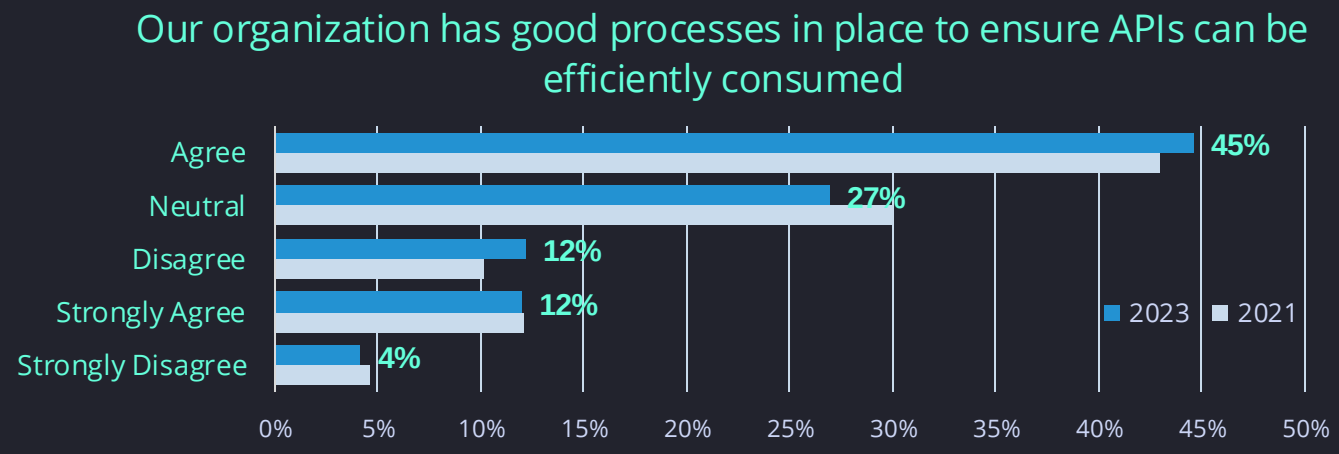
SMARTBEAR

Alignment Gap



› Only 48% are confident their APIs are well documented and explain value for consumers

› ~43% unsure their processes will deliver appropriate developer experience (DX)



Source: SmartBear – State of Software Quality | API 2023

SMARTBEAR

CAUTION:
TREES
OBSCURING
VIEW OF
FOREST
NEXT 5 MILES

Arazzo

What is the Arazzo Specification?

- It describes an approach to document use-case oriented workflows in a programmatically readable format.
- A workflow is a series of API calls which when woven 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.

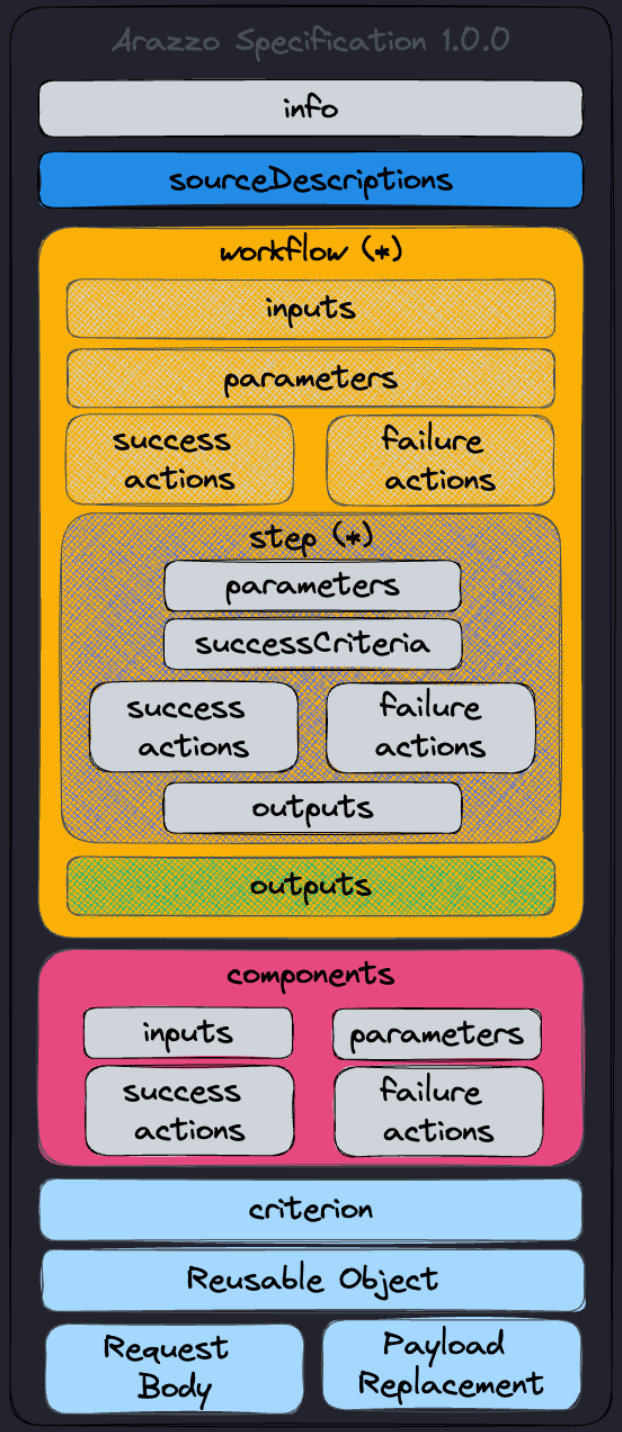
SMARTBEAR

Arazzo

Use cases

- Deterministic recipes for the use of APIs
  - Make sense of large unwieldy API description
  - Bridge the *gap* where business flows span more than one API description
- Living API documentation and improved consumer developer experience (DX)
- Assertable business value
- Targeted code generation tooling for APIs driven by use cases
- Improved regulatory checks via assertable workflows
- The AI potential for value orientated interaction with APIs

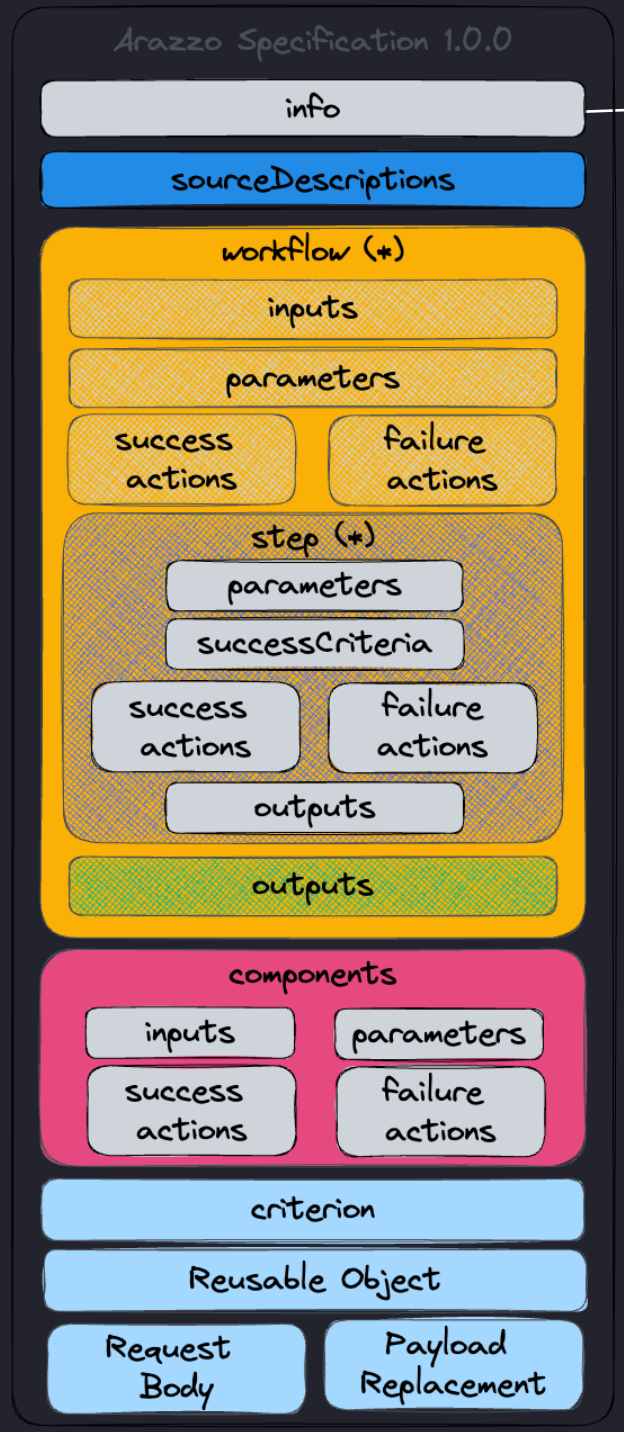
SMARTBEAR



Arazzo

Specification Structure

SMARTBEAR



Metadata about the defined Arazzo Document

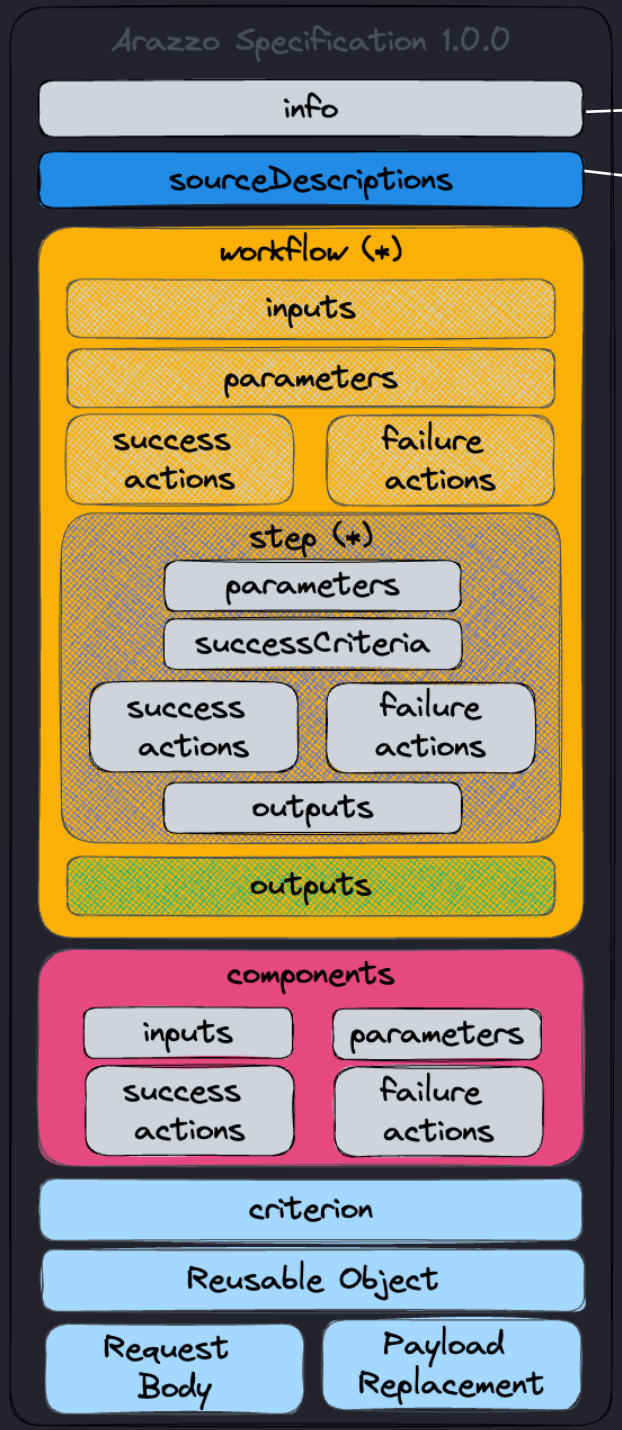
SMARTBEAR

Metadata about the defined Arazzo Document

Arazzo Specification - Info Object Example

```yaml
arazzo: 1.0.0
info:
  title: A pet adoption workflow
  summary: Search for and adopt a pet from the PetCo
  description: This workflow walks you through the steps of `searching` for, `selecting`,
and `adopting` an available pet.
  version: 1.1.0
```

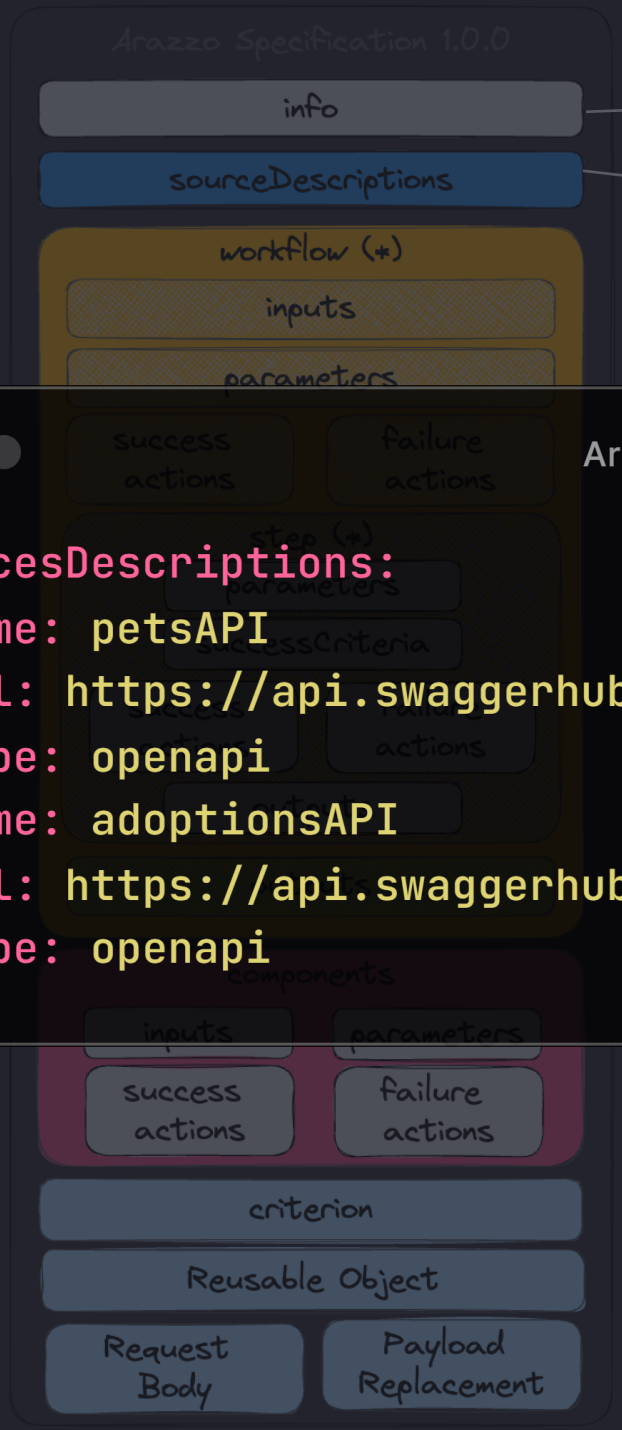
SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

SMARTBEAR



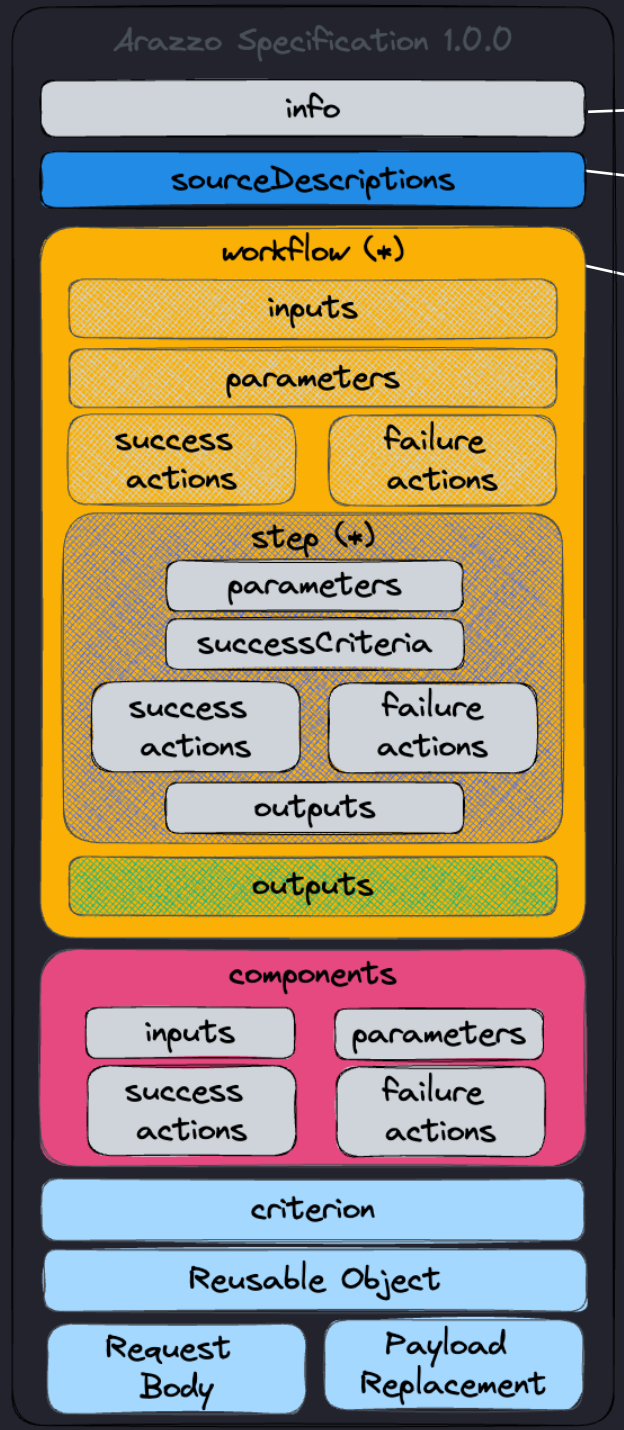
Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Arazzo Specification - Source Descriptions Example

```yaml
sourcesDescriptions:
- name: petsAPI
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Pets-API/1.0.0/swagger.json
  type: openapi
- name: adoptionsAPI
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Adoptions-API/1.0.0/swagger.json
  type: openapi
```

SMARTBEAR

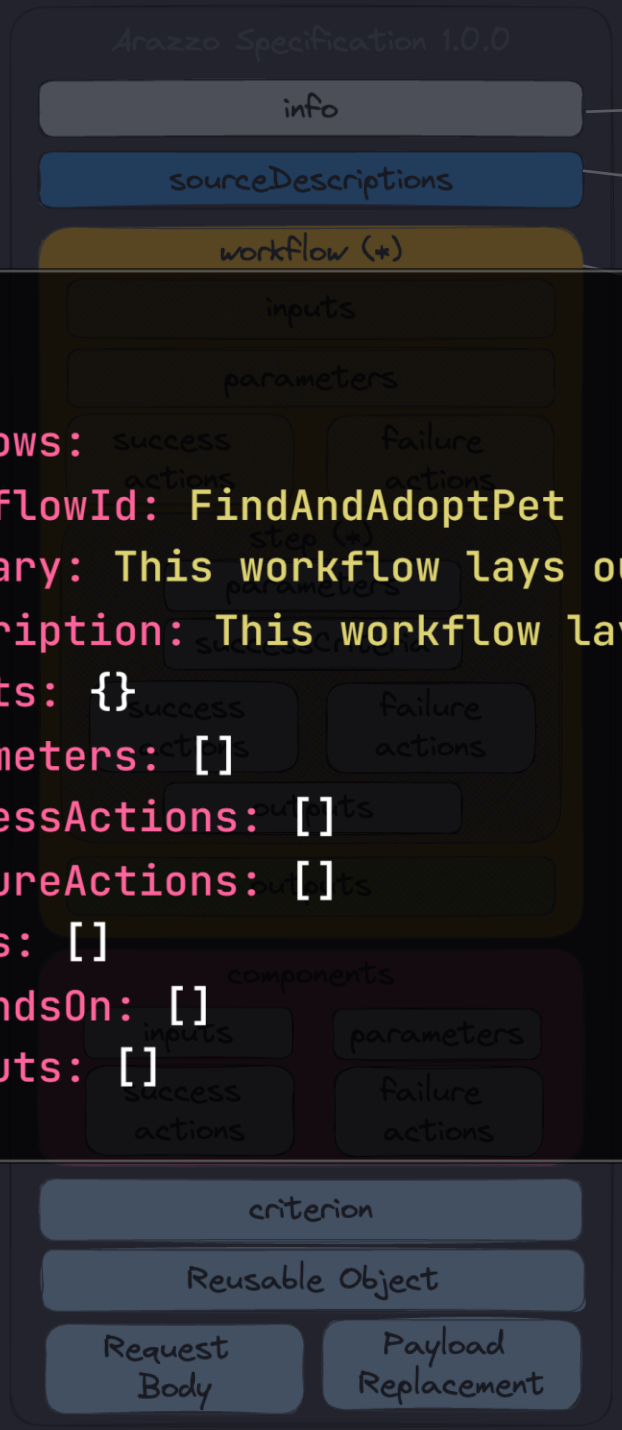


Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

SMARTBEAR



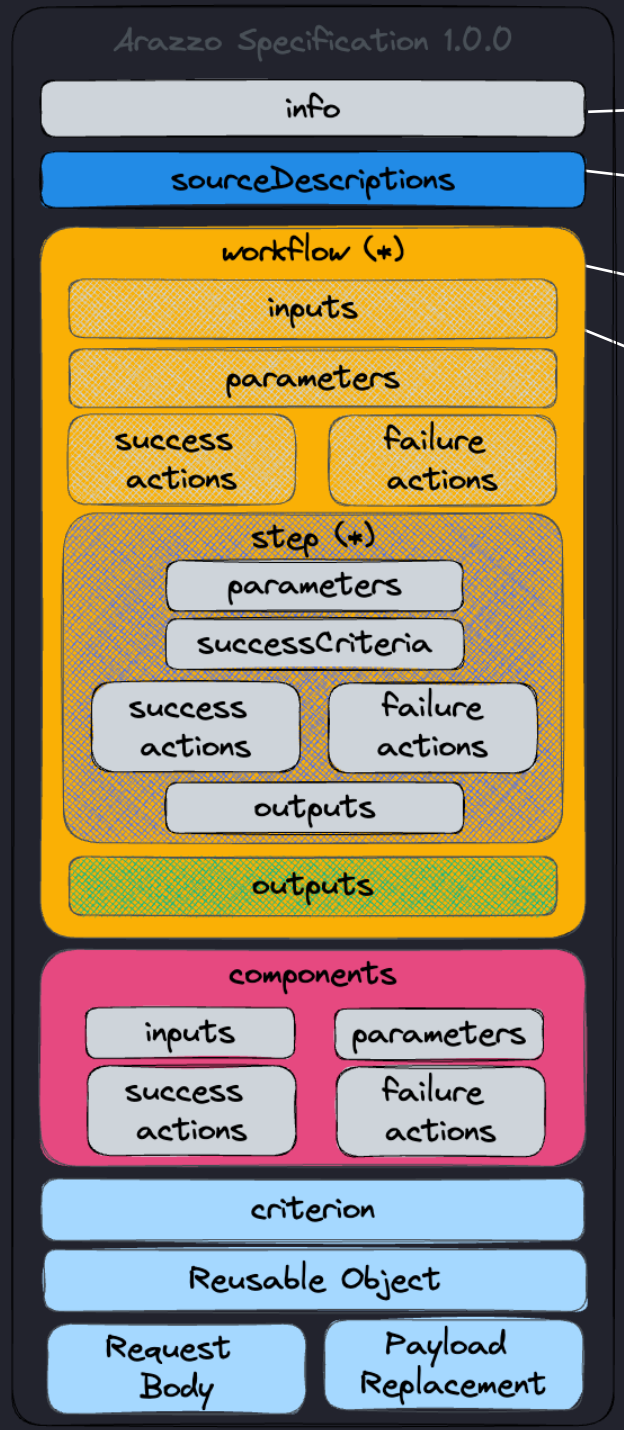
Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Arazzo Specification -Workflows Example

```yaml
workflows:
- workflowId: FindAndAdoptPet
  summary: This workflow lays out the steps and API calls needed to search for and adopt a Pet
  Description: This workflow lays out the steps needed to adopt a pet.
  inputs: {}
  parameters: []
  successActions: []
  failureActions: []
  steps: []
  dependsOn: []
  outputs: []
```

SMARTBEAR



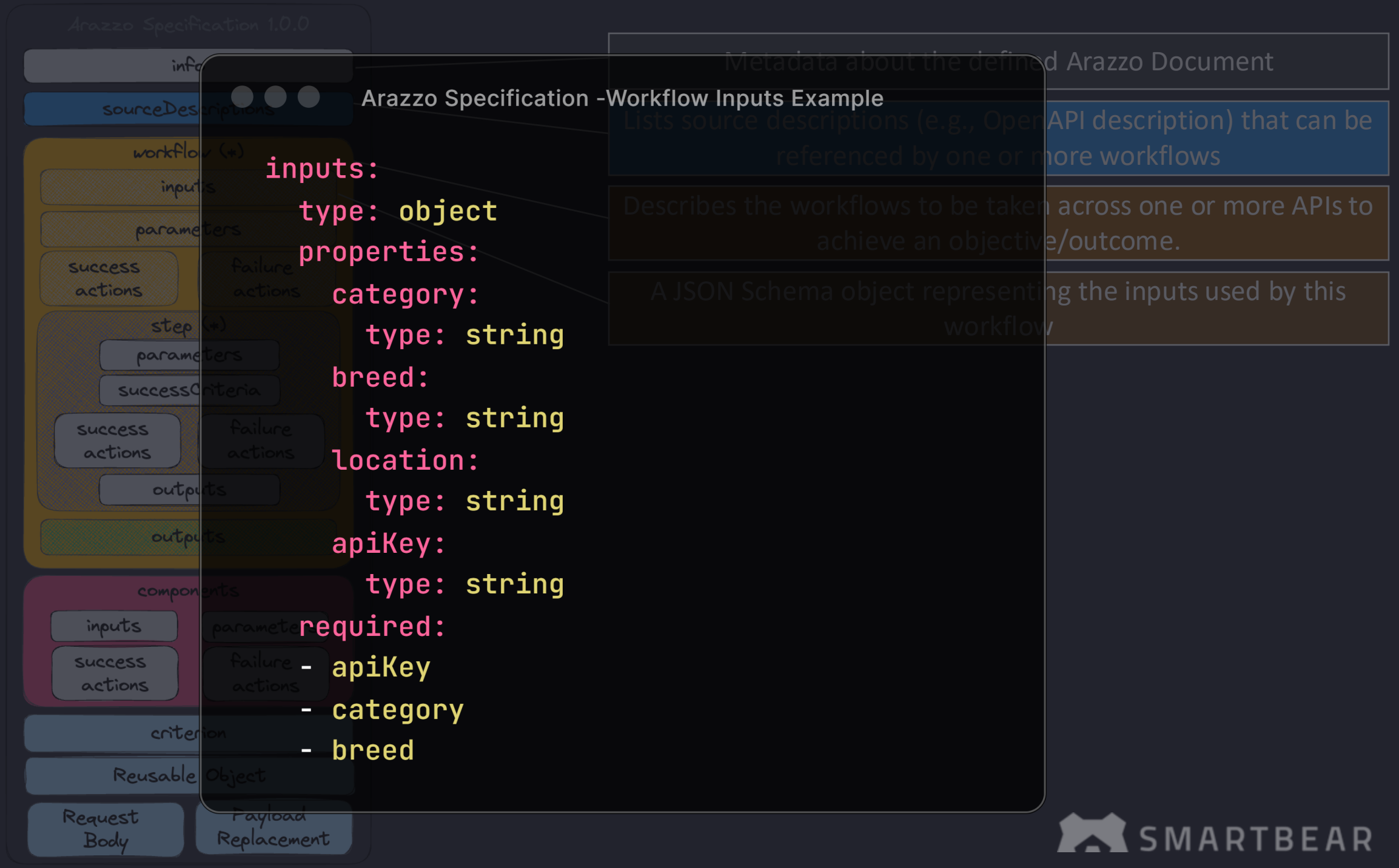
Metadata about the defined Arazzo Document

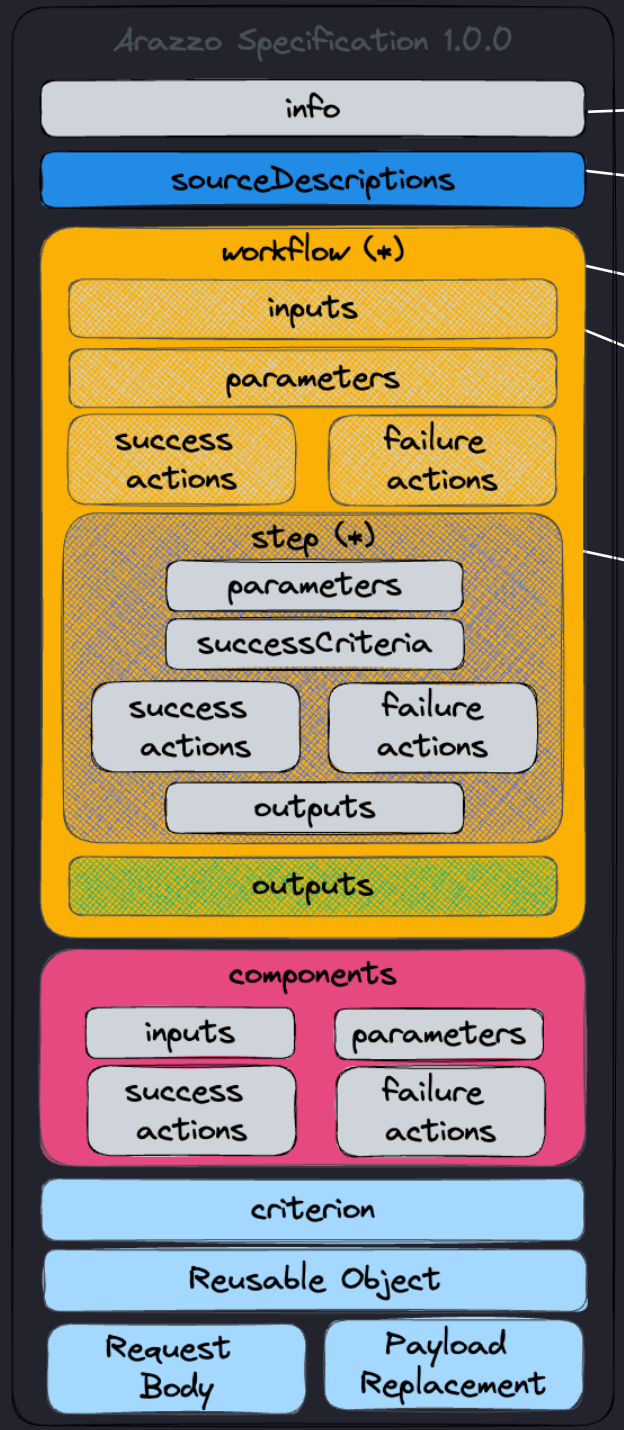
Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

SMARTBEAR


Arazzo Specification 1.0.0
info
sourceDescriptions
workflow (+)
inputs
parameters
success actions
failure actions
step (+)
parameters
successCriteria
success actions
failure actions
outputs
outputs
components
inputs
parameters
success actions
failure actions
criterion
Reusable Object
Request Body
Payload Replacement
Arazzo Specification -Workflow Inputs Example
inputs:
type: object
properties:
category:
type: string
breed:
type: string
location:
type: string
apiKey:
type: string
required:
- apiKey
- category
- breed
Metadata about the defined Arazzo Document
Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows
Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.
A JSON Schema object representing the inputs used by this workflow
SMARTBEAR



Metadata about the defined Arazzo Document

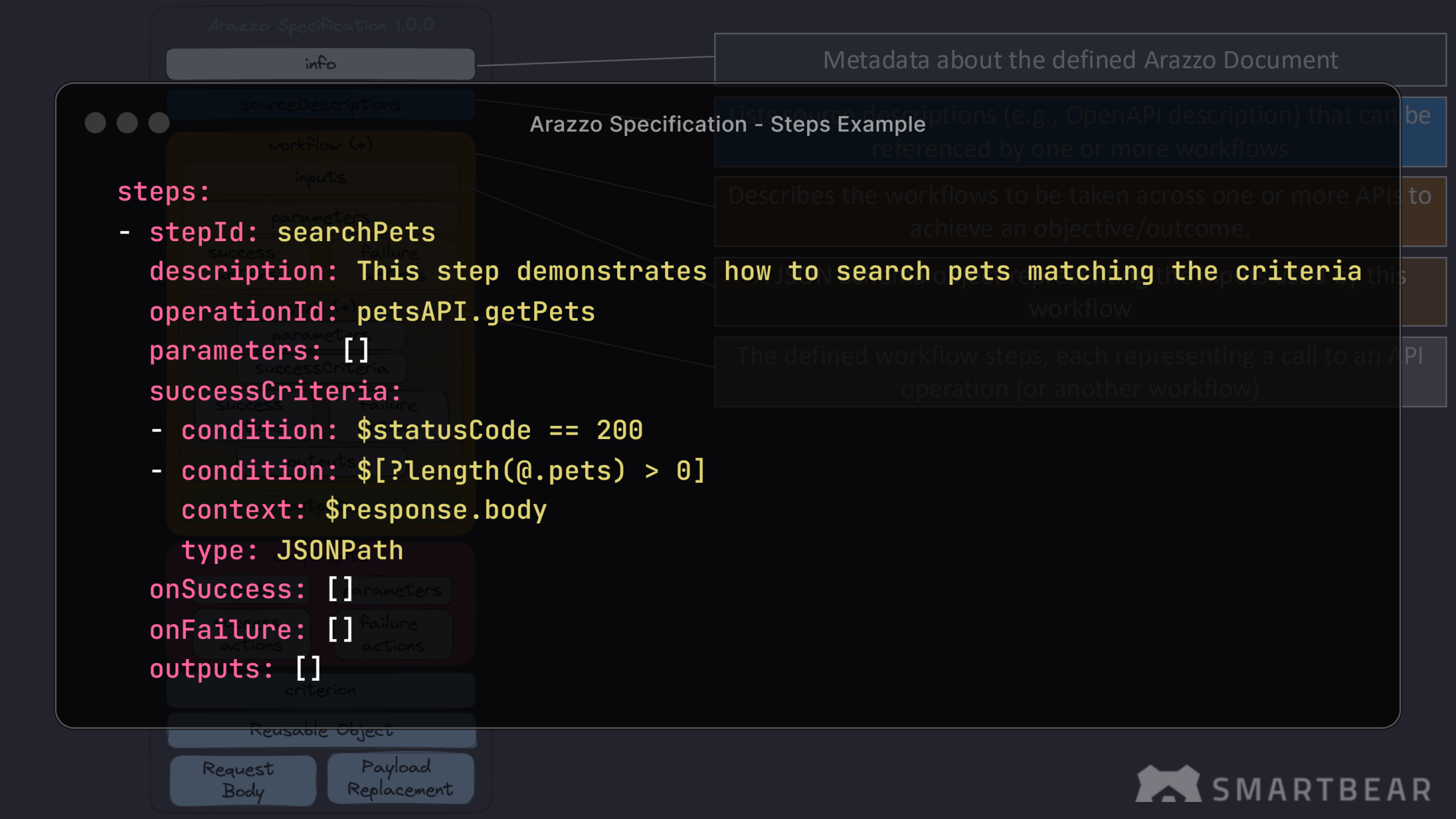
Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

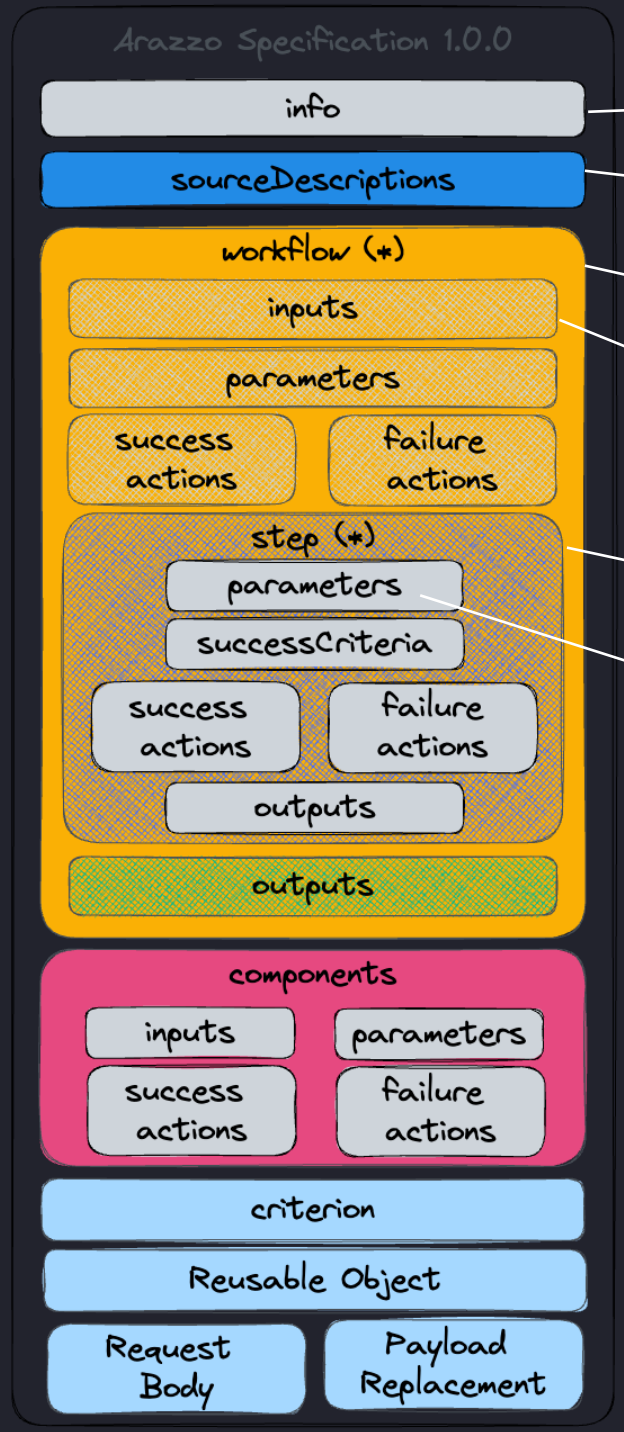
Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

SMARTBEAR


Arazzo Specification - Steps Example
steps:
- stepId: searchPets
description: This step demonstrates how to search pets matching the criteria
operationId: petsAPI.getPets
parameters: []
successCriteria:
- condition: $statusCode == 200
- condition: $[?length(@.pets) > 0]
context: $response.body
type: JSONPath
onSuccess: []
onFailure: []
outputs: []
Metadata about the defined Arazzo Document
List source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows
Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.
The defined workflow steps, each representing a call to an API operation (or another workflow)



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

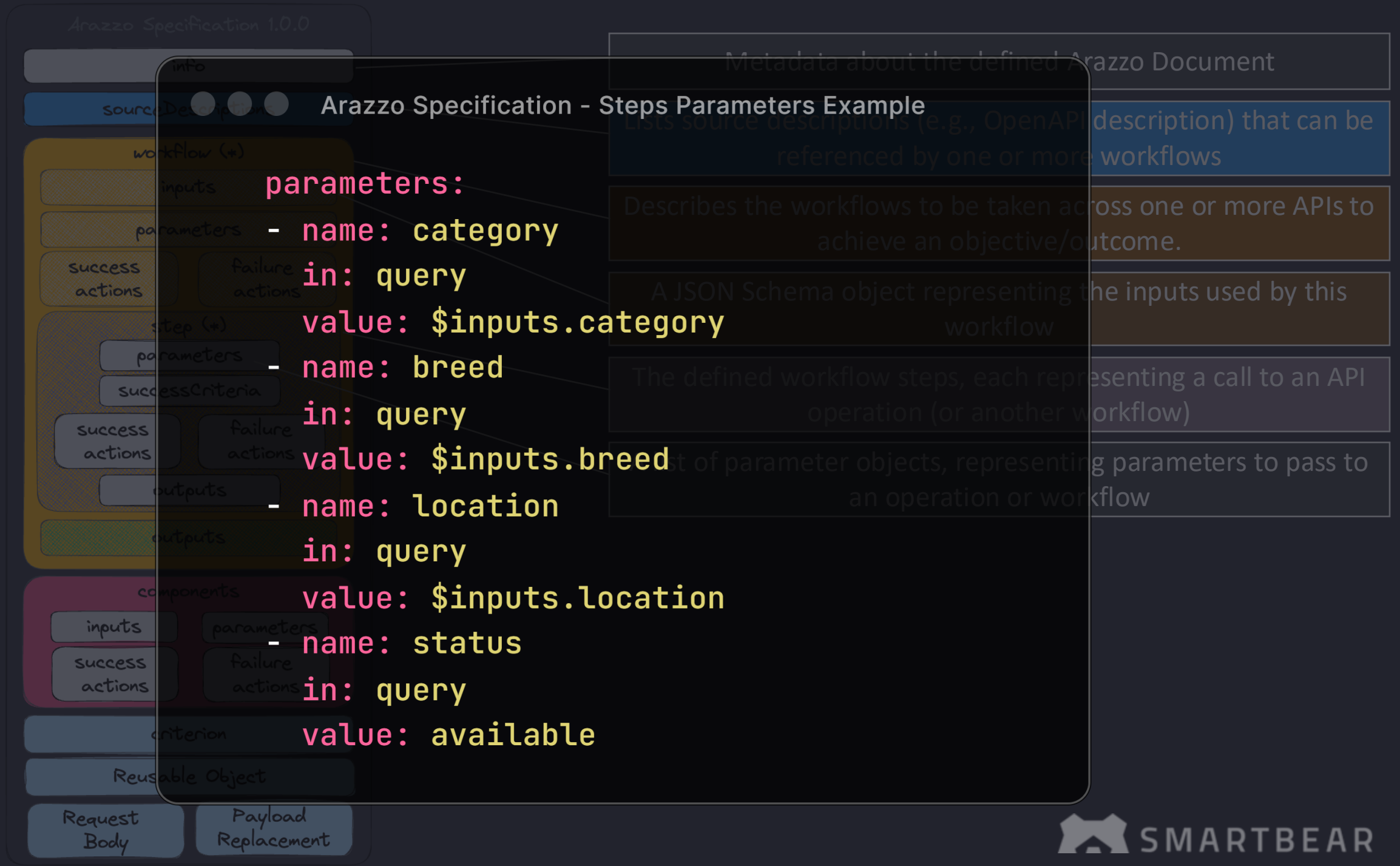
Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

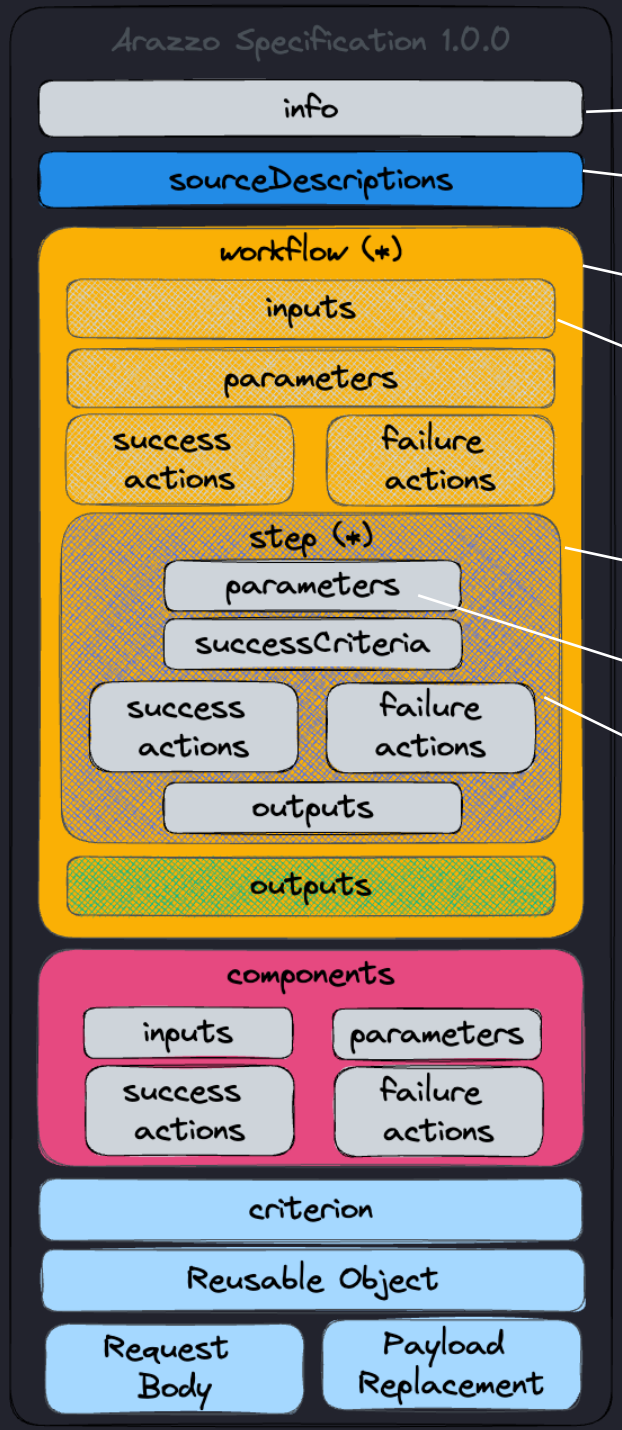
A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

SMARTBEAR


Arazzo Specification 1.0.0
info
source
workflow (+)
inputs
parameters
success actions
failure actions
step (+)
parameters
successCriteria
success actions
failure actions
outputs
outputs
components
inputs
parameters
success actions
failure actions
criterion
Reusable Object
Request Body
Payload Replacement
Arazzo Specification - Steps Parameters Example
parameters:
- name: category
in: query
value: $inputs.category
- name: breed
in: query
value: $inputs.breed
- name: location
in: query
value: $inputs.location
- name: status
in: query
value: available
Metadata about the defined Arazzo Document
Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows
Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.
A JSON Schema object representing the inputs used by this workflow
The defined workflow steps, each representing a call to an API operation (or another workflow)
List of parameter objects, representing parameters to pass to an operation or workflow
SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

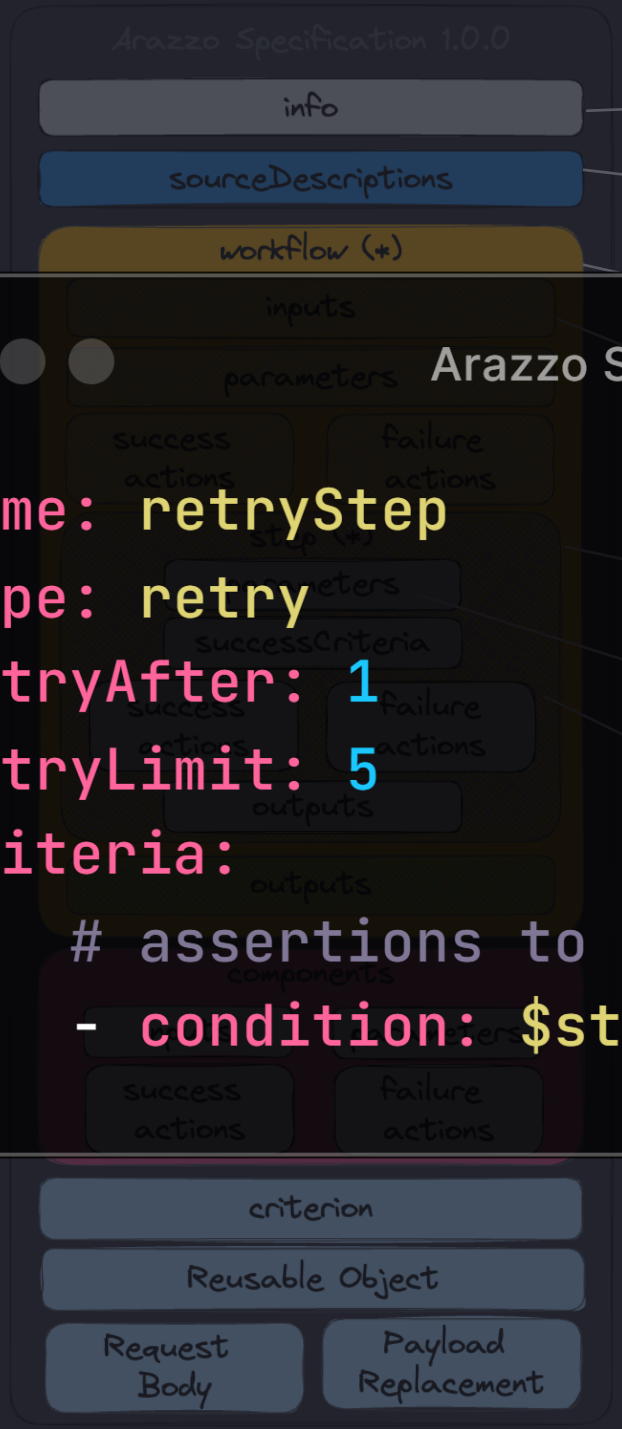
A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

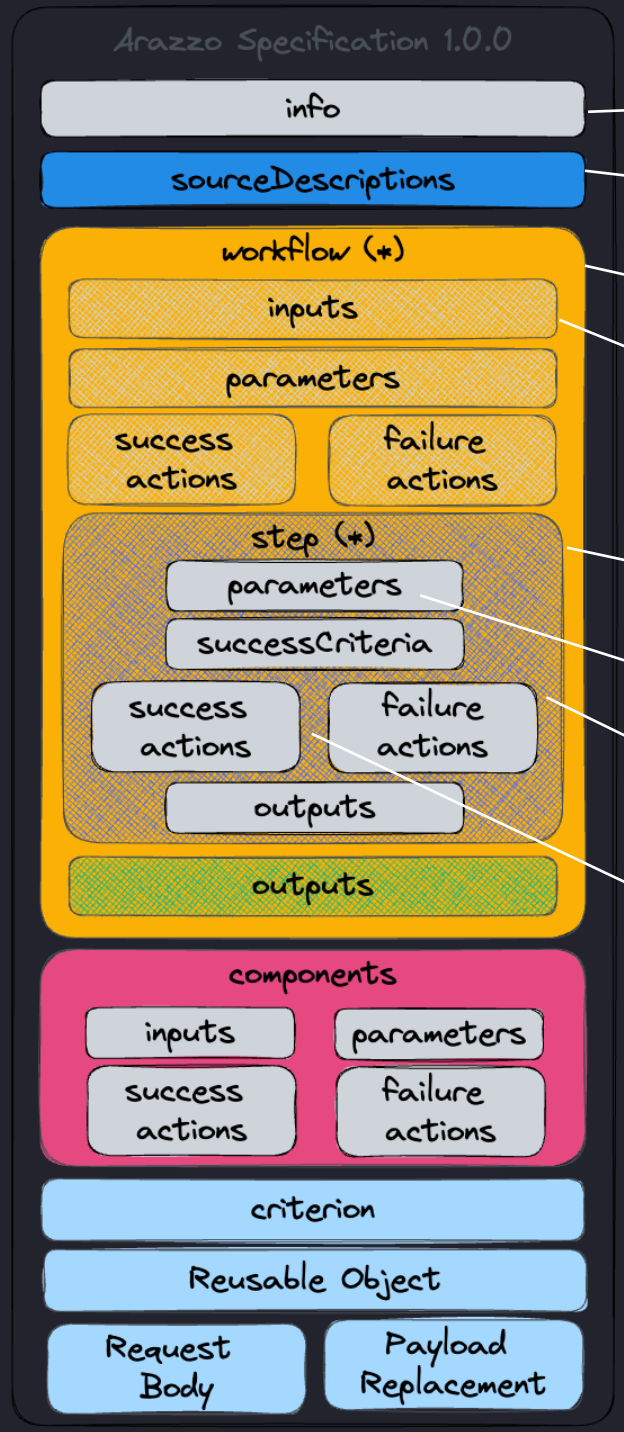
A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

Arazzo Specification - FailureAction Object Example

```yaml
name: retryStep
type: retry
retryAfter: 1
retryLimit: 5
criteria:
    # assertions to determine if this action should be executed
    - condition: $statusCode == 503
```

SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

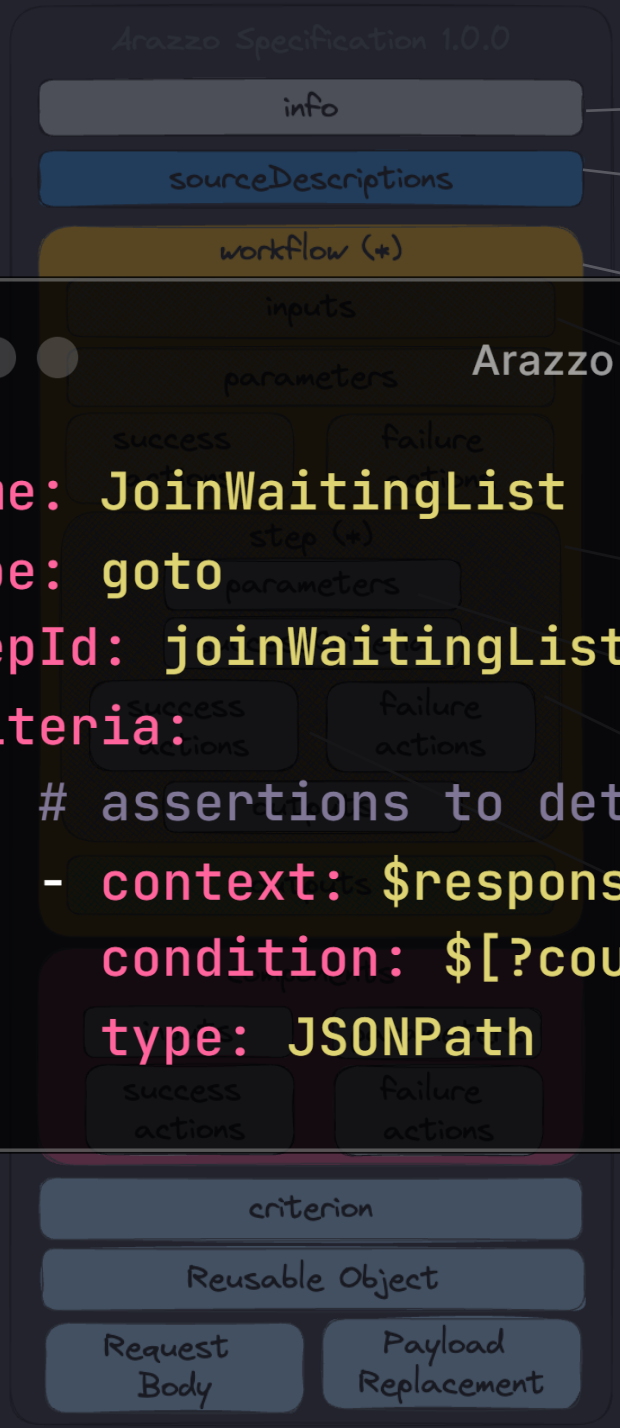
The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

An array of success action objects that specify what to do upon step success

SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

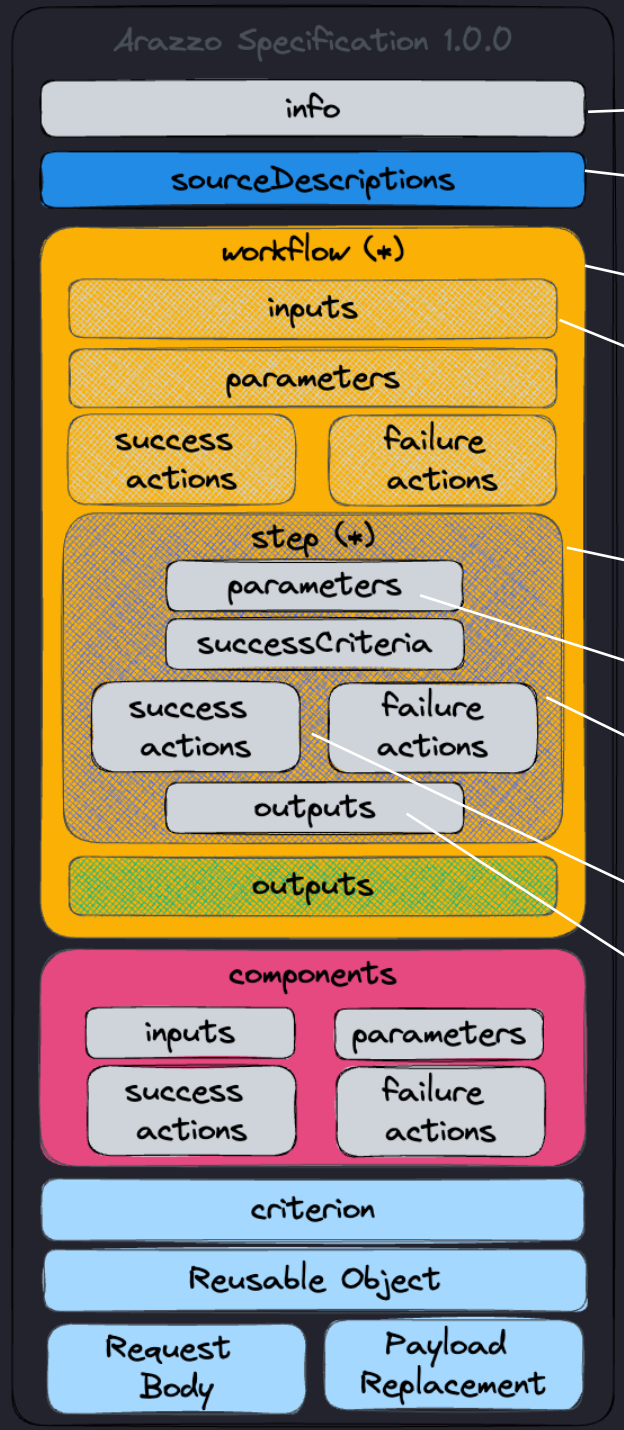
An array of failure action objects that specify what to do on step failure

An array of success action objects that specify what to do upon step success

Arazzo Specification - SuccessAction Object Example

```yaml
name: JoinWaitingList
type: goto
stepId: joinWaitingListStep
criteria:
    # assertions to determine if this success action should be executed
    - context: $response.body
      condition: $[?count(@.pets) > 0]
      type: JSONPath
```

SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

An array of success action objects that specify what to do upon step success

A map between a friendly name and a dynamic output value for a step

SMARTBEAR

Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

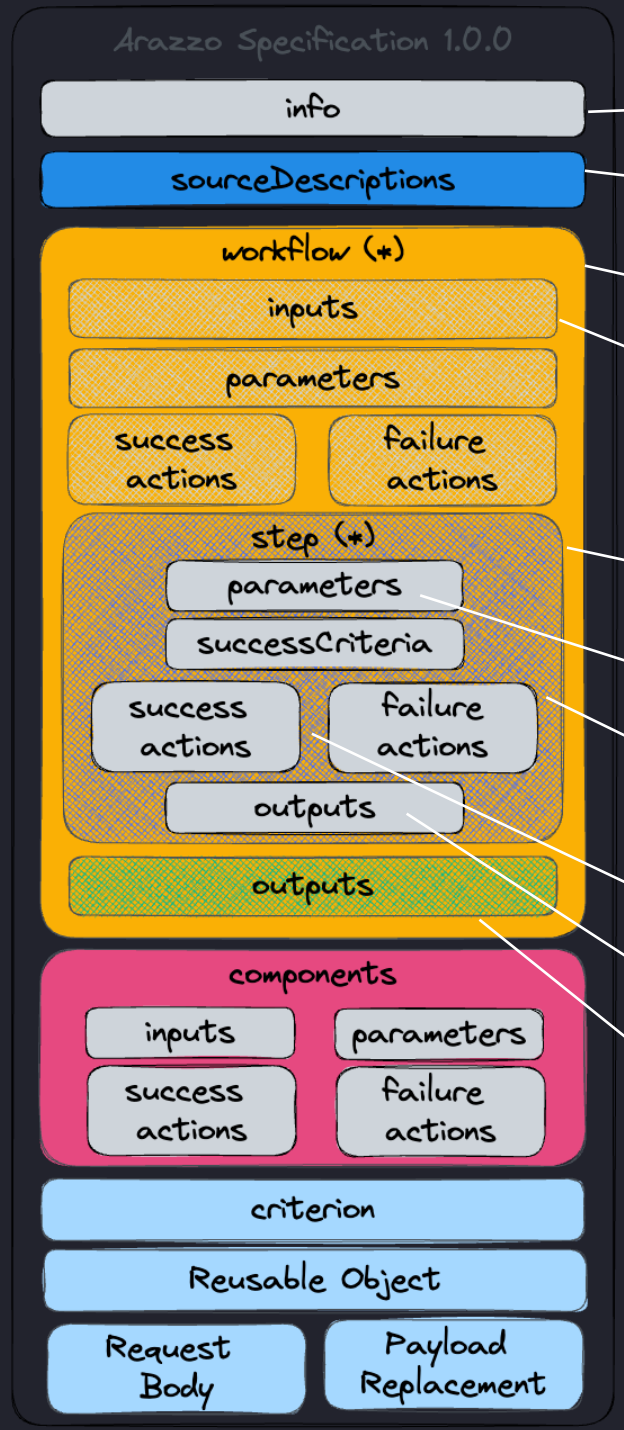
An array of success action objects that specify what to do upon step success

A map between a friendly name and a dynamic output value for a step

Arazzo Specification - Step Outputs Example

```
outputs:
  petId: $response.body.pets[0].id
```

SMARTBEAR



Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A JSON Schema object representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

An array of success action objects that specify what to do upon step success

A map between a friendly name and a dynamic output value for a step

A map between a friendly name and a dynamic output value for a workflow

Metadata about the defined Arazzo Document

Lists source descriptions (e.g., OpenAPI description) that can be referenced by one or more workflows

Describes the workflows to be taken across one or more APIs to achieve an objective/outcome.

A list of parameter objects representing the inputs used by this workflow

The defined workflow steps, each representing a call to an API operation (or another workflow)

A list of parameter objects, representing parameters to pass to an operation or workflow

An array of failure action objects that specify what to do on step failure

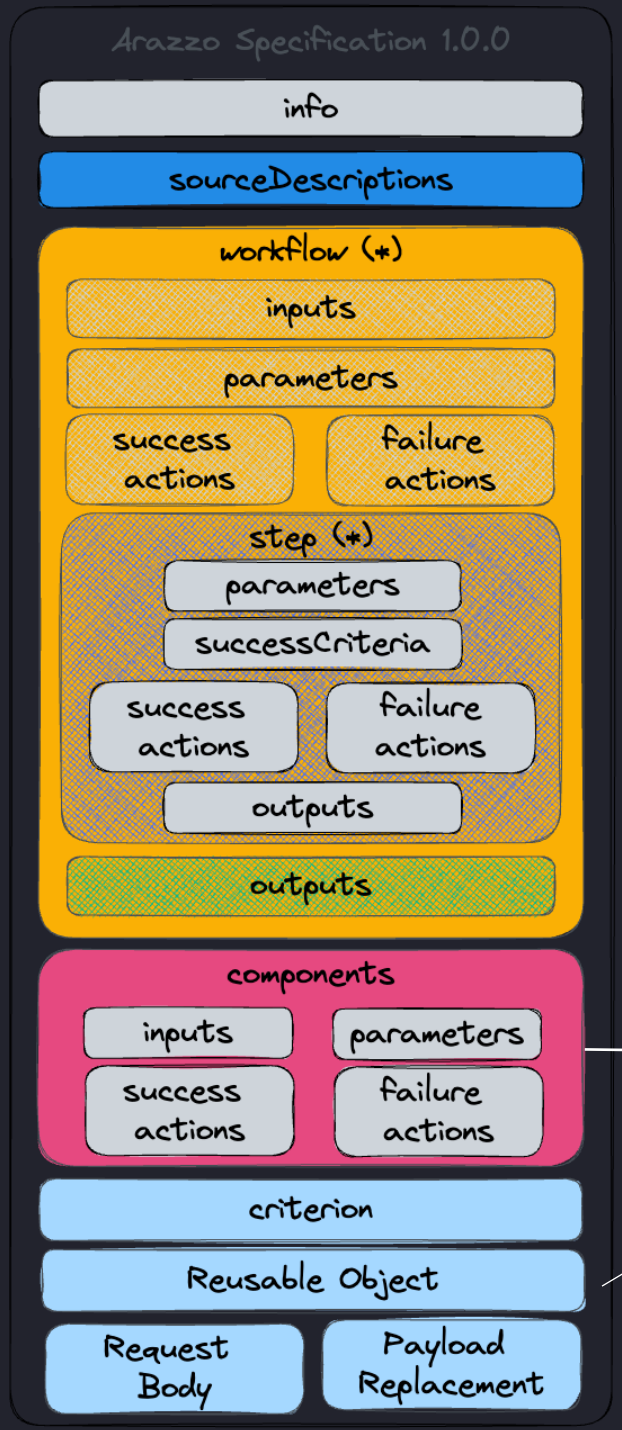
An array of success action objects that specify what to do upon step success

A map between a friendly name and a dynamic output value for a step

A map between a friendly name and a dynamic output value for a workflow

Arazzo Specification - Workflow Outputs Example

```yaml
outputs:
  petId: $steps.searchPets.outputs.petId
  petName: $steps.searchPets.outputs.petName
  status: $steps.approveAdoption.outputs.status
```



Reusable (referenceable) components and objects

SMARTBEAR

BNPL by Free Debt Student Bank

Problem:
Retailers are preparing for the student rush, but many students struggle to afford essential study equipment, such as laptops and software, without incurring immediate debt.

Solution:
To ease the financial burden on students, Free Debt Student Bank has launched a BNPL service designed specifically for students. This service allows retailers to offer flexible, student-friendly payment plans at checkout, enabling students to buy necessary equipment with minimal upfront costs and 0% loan rates.

Retailers can seamlessly integrate this service to attract and support students during the busy back-to-school season.

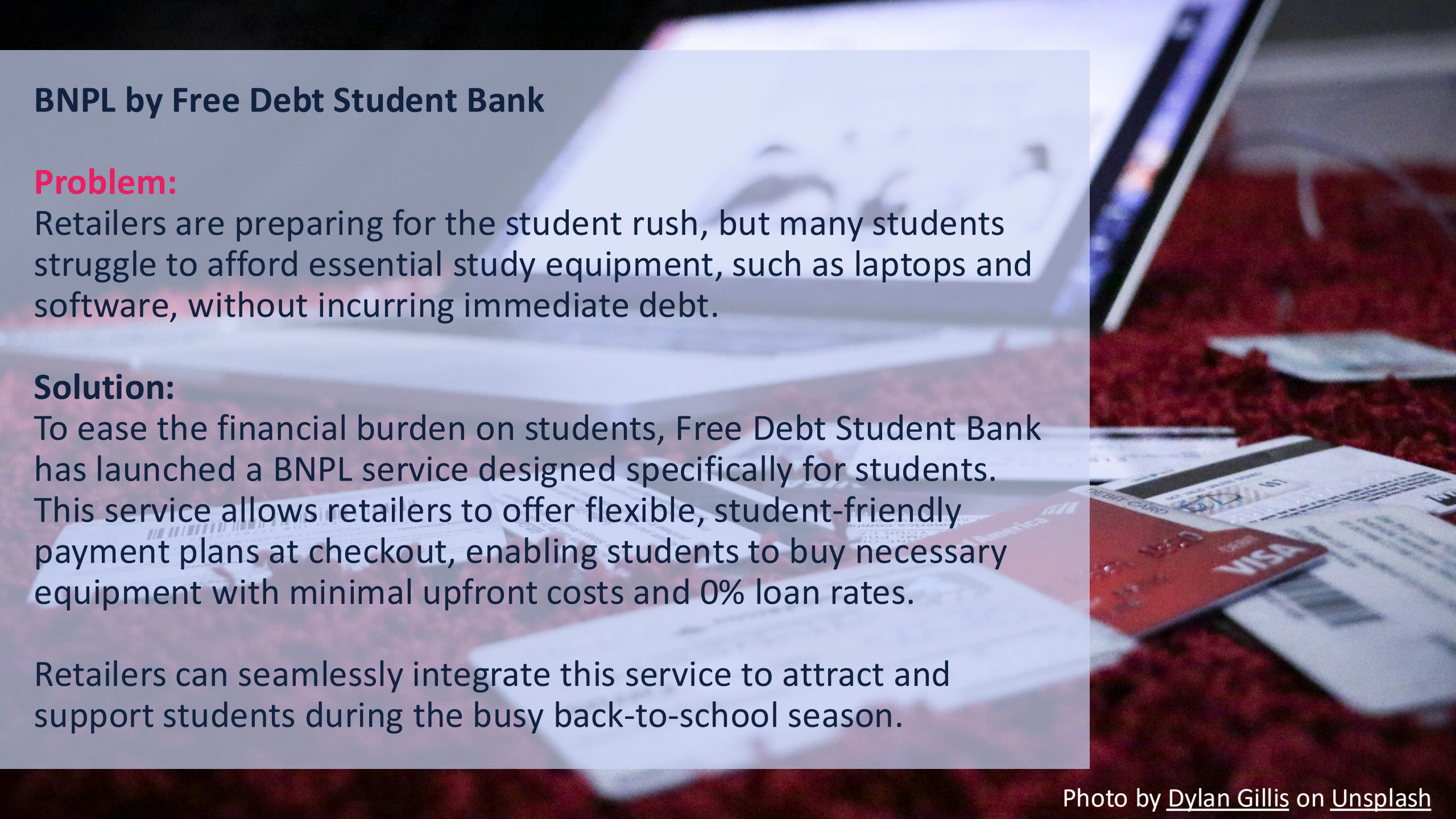


Photo by Dylan Gillis on Unsplash

Example Workflow Use Case

Arazzo

› Enable a Buy Now Pay Later Checkout Option for Online Products

› Steps:

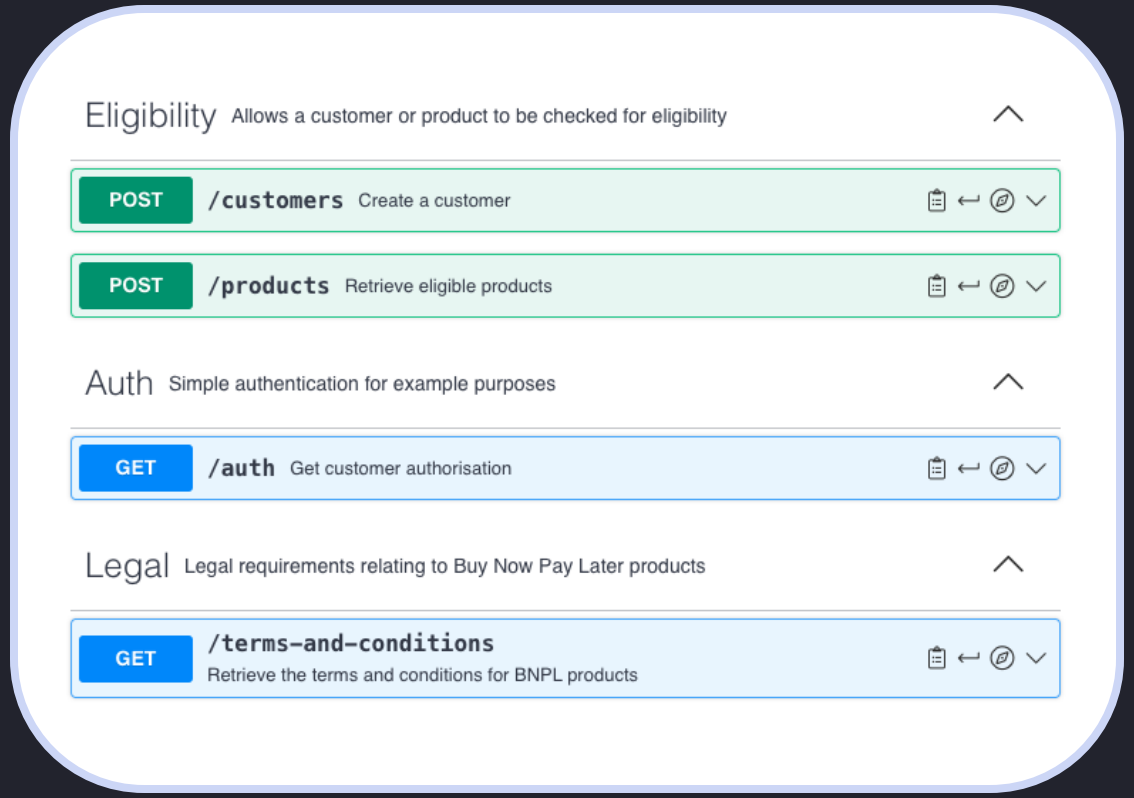
1. Check selected Products are BNPL eligible
2. Retrieve T&Cs and determine Customer eligibility
3. Create customer record if needed
4. Initiate BNPL Loan Transaction
5. Authenticate Customer and obtain Authorization for Loan
6. Calculate and retrieve Payment Plan to show within client app
7. Update Order Status

SMARTBEAR

Multiple APIs and Endpoints Required

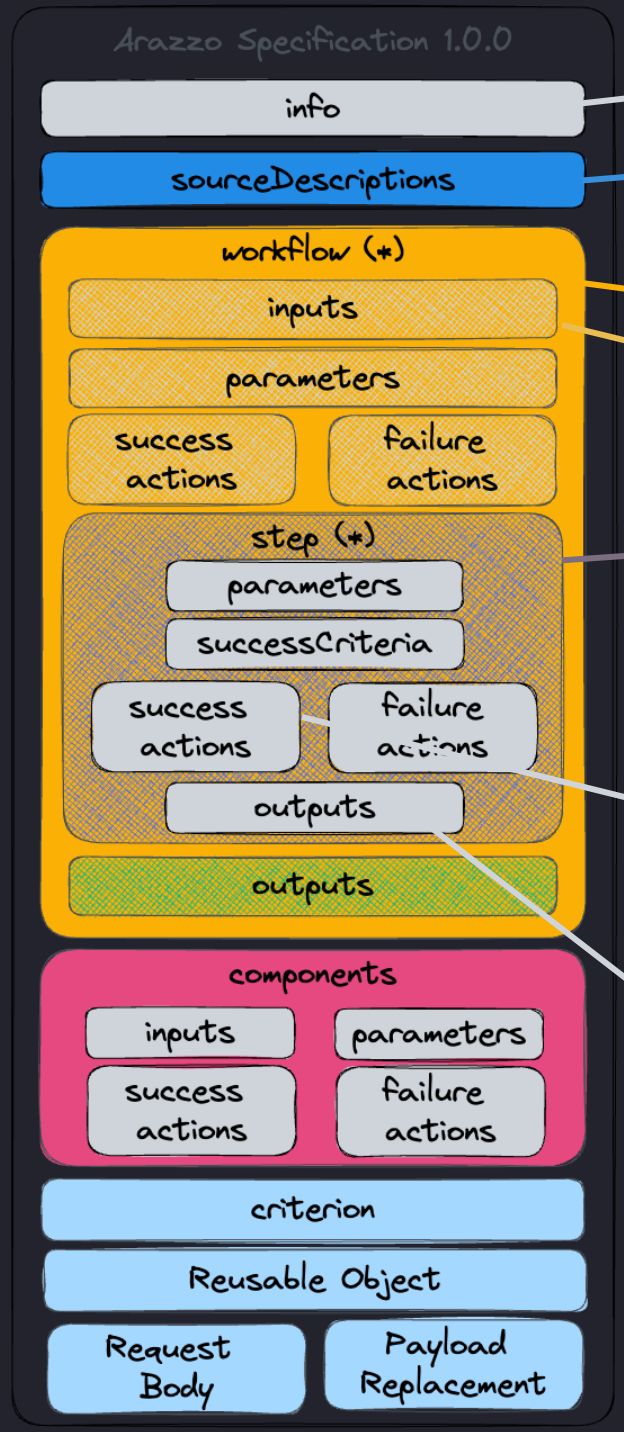
Arazzo

Buy Now, Pay Later - Eligibility API



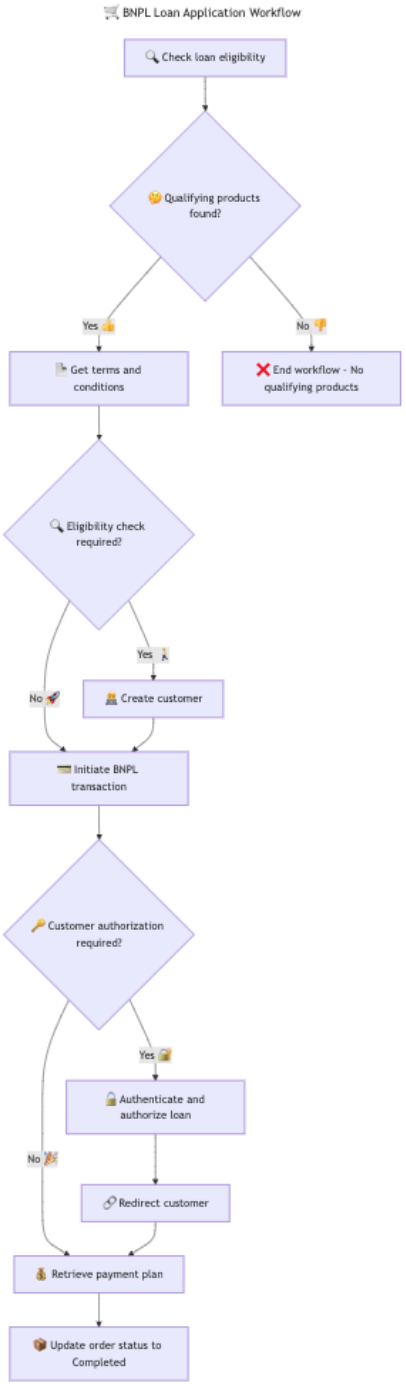
Buy Now, Pay Later - Loan API

SMARTBEAR

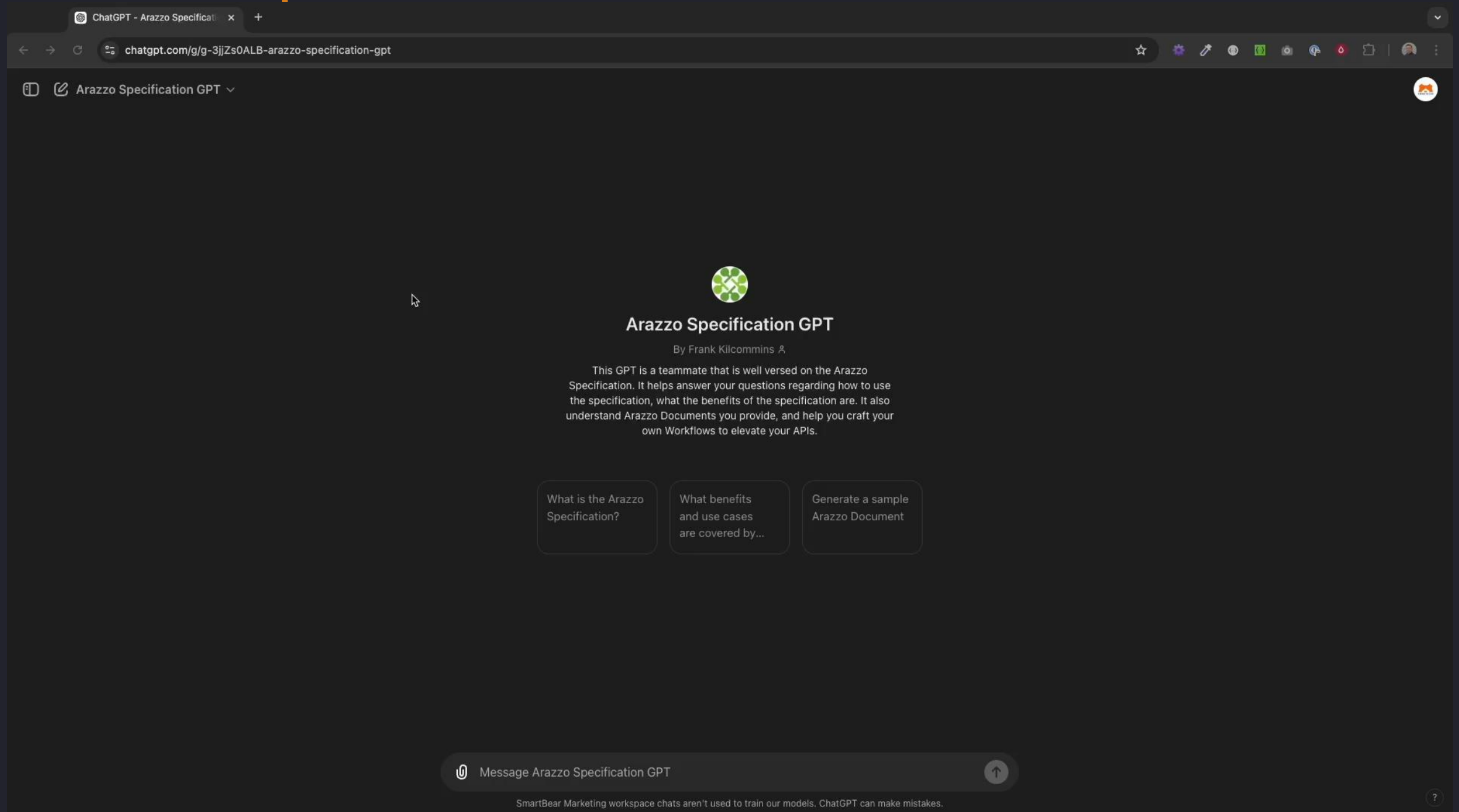



```yaml
arazzo: 1.0.0
info:
  title: BNPL Workflow description
  version: 1.0.0
sourceDescriptions:
  - name: BnplEligibilityApi
    url: https://api.swaggerhub.com/apis/frank-kilcommins/BNPL-Eligibility-and-Auth/1.0.0/swagger.json
    type: openapi
  - name: BnplLoanApi
    url: https://api.swaggerhub.com/apis/frank-kilcommins/BNPL-Loans/1.0.0/swagger.json
    type: openapi
workflows:
- workflowId: ApplyForLoanAtCheckout
  summary: Apply for a loan at checkout using a BNPL platform
  description: Describes the steps to secure a loan at checkout from a BNPL platform.
  inputs: …
  steps:
  - stepId: checkLoanCanBeProvided
    description: |
      Call the BNPL API to filter the basket for products qualifying for checkout loans.
    operationId: $sourceDescriptions.BnplEligibilityApi.findEligibleProducts
    requestBody:
      contentType: application/json
      payload: |
        {
          "customer": "{$inputs.customer}",
          "products": "{$inputs.products}"
        }
    successCriteria:
      - condition: $statusCode == 200
    onSuccess:
      - name: existingCustomerNotEligible
        type: end
        criteria:
          - condition: $statusCode == 200
          - condition: $response.body.existingCustomerNotEligible == false
      - name: qualifyingProductsFound …
      - name: qualifyingProductsNotFound
        type: end
        criteria:
          - condition: $statusCode == 200
          - context: $response.body
            type: jsonpath
            condition: $[?count(@.products) == 0]
    outputs:
      eligibilityCheckRequired: $response.body.eligibilityCheckRequired
      eligibleProducts: $response.body.products
      totalLoanAmount: $response.body.totalAmount
```

```yaml
arazzo: 1.0.0
info:
  title: A pet adoption workflow
  summary: Search for and adopt a pet from the PetCo
  description: This workflow walks you through the steps of `searching` for, `selecting`, and `adopting` a
  version: 1.1.0
sourcesDescriptions:
- name: petsAPI
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Pets-API/1.0.0/swagger.json
  type: openapi
- name: adoptionsAPI
  url: https://api.swaggerhub.com/apis/frank-kilcommins/Adoptions-API/1.0.0/swagger.json
  type: openapi
workflows:
- workflowId: FindAndAdoptPet
  summary: This workflow lays out the steps and API calls needed to search for and adopt a Pet
  inputs:
    type: object
    properties:
      category:
        type: string
      breed:
        type: string
      location:
        type: string
      apiKey:
        type: string
    required:
    - apiKey
    - category
    - breed
  steps:
  - stepId: searchPets
    description: This step demonstrates the search pets flow for the first pet matching the criteria
    operationId: petsAPI.getPets
    parameters:
    - name: category
      in: query
      value: $inputs.category
    - name: breed
      in: query
      value: $inputs.breed
    - name: location
      in: query
      value: $inputs.location
    - name: status
      in: query
      value: available
    successCriteria:
    - context: $response.body
      condition: $[?length(@.pets) > 0]
      type: JSONPath
    outputs:
      petId: $response.body.data.pets[0].id
      petName: $response.body.data.pets[0].name
      petLocation: $response.body.data.pets[0].location
```



Arazzo Specification GPT



Emerging Tooling

Arazzo

liblab

Redocly

APIMATIC

SPEAKEASY

SMARTBEAR

Stoplight
A SMARTBEAR COMPANY

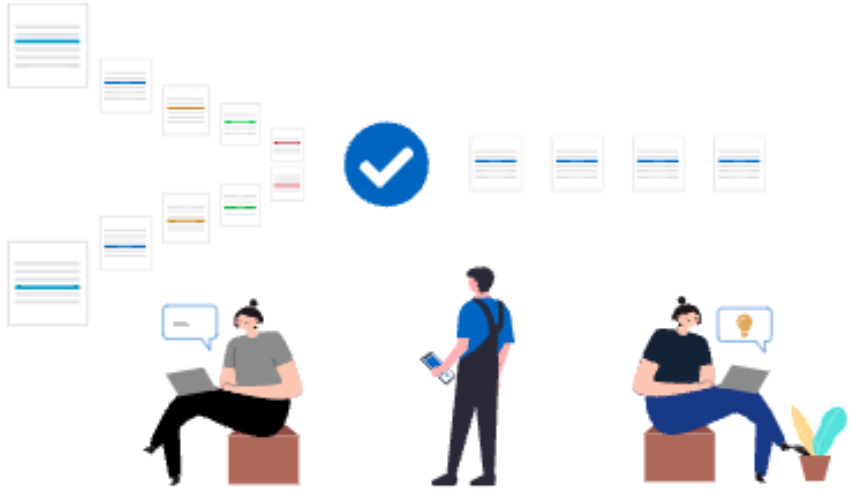
Spectral Arazzo Support

Arazzo

SPECTRAL
Brought to you by **Stoplight**.

JSON/YAML Linter with **Custom Rulesets**

Spectral is a JSON/YAML linter with out of the box support for **OpenAPI 3.x & 2.0**, **Arazzo** and **AsyncAPI**



Just add "spectral:arazzo" to your ruleset

```
extends:
 - "spectral:arazzo"
```

SMARTBEAR

Calls to Action

- Create an Arazzo Description and lint it with Spectral
- Join the slack workspace: https://communityinviter.com/apps/open-api/openapi
- Star the repo: https://github.com/OAI/Arazzo-Specification
- Submit your use cases (as discussions or issues)

Learn more

- Arazzo Specification + Examples:
  - https://github.com/OAI/Arazzo-Specification

Arazzo
SMARTBEAR

› Connect:

@fkilcommins

@frank-kilcommins

frank.kilcommins@smartbear.com

Q&A

Thanks

FS9U

ProxyHands

LangDev CON 2024

Seville 17-19 October, 2024

https://langdevcon.org