

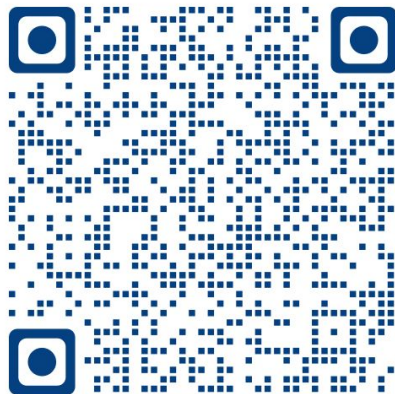
# The API Product Canvas

## How to Design OpenAPIs in Workshops: The API Product Canvas

Dr. Annegret Junker &  
Fabrizio Lazzaretti

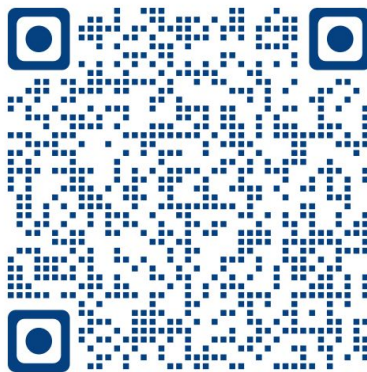


July 2nd & 3rd, 2025



Architect at codecentric

Interested in Microservices, DDD, API Design,  
Event Design and how to get faster from  
specification to code



Architect at Wavestone

interested in API Design, Event Design,  
Microservices and how to make a developer's life  
easier

# Problem

1	openapi: 3.0.3	69	description: success.	139	required: true	221	type: number
2	info:	70	content:	140	schema:	222	description: The budget amount associated with the portfolio. Cannot be 'null'.
3	version: '2.0'	71	application/json:	141	type: number	223	currencyCode:
4	title: Amazon Ads API - Portfolios	72	schema:	142	responses:	224	type: string
5	description:  -	73	items:	143	'200':	225	description:  -
6	Portfolios consist of campaigns that are grouped together and link	74	type: object	144	description: success.	226	The currency used for all monetary values for entities under this profile. Canno
7	contact:	75	code:	145	content:	227	[Region] countryCode [country name] currencyCode
8	name: Amazon Ads API Support	76	properties:	146	application/json:	228	----- ----- -----
9	url: https://amzn-clicks.atlassian.net/servicesdc/customer/portal	77	description: a string, 'SUCCESS', that indicates the por	147	schema:	229	[NA]US[United States]USD
10	license:	78	type: string	148	\$ref: '#/components/schemas/Portfolio'	230	[CA]CA[Canada]CAD
11	name: Amazon Ads API License Agreement	79	portfolioId:	149	'401': '#/components/responses/unauthorized'	231	[MX]MX[Mexico]MXN
12	url: https://advertising.amazon.com/API/docs/license_agreement	80	description: The portfolio identifier.	150	'404': '#/components/responses/notfound'	232	[BR]BR[Brazil]BRL
13	servers:	81	type: number	151	\$ref: '#/components/responses/notfound'	233	[UK]UK[United Kingdom]GBP
14	- url: https://advertising-api.amazon.com	82	'401':	152	\$ref: '#/components/responses/notfound'	234	[DE]DE[Germany]EUR
15	- description: Production endpoint for North America, which includes	83	\$ref: '#/components/responses/unauthorized'	153	/\$v2/portfolios/extended:	235	[FR]FR[France]EUR
16	- url: https://advertising-api-eu.amazon.com	84	'404':	154	get:	236	[ES]ES[Spain]EUR
17	- description: Production endpoint for Europe, which includes UK, FR,	85	\$ref: '#/components/responses/unauthorized'	155	Scan   Try   Audit	237	[IT]IT[Italy]EUR
18	- url: https://advertising-api-fe.amazon.com	86	\$ref: '#/components/responses/notfound'	156	tags:	238	[NL]NL[Netherlands]EUR
19	- description: Production endpoint for Far East, which includes JP AR	87	post:	157	- Portfolios extended	239	[SE]SE[Sweden]SEK
20	tags:	88	Scan   Try   Audit	158	summary: Gets a list of portfolios with an extended set of properties.	240	[PL]PL[Poland]PLN
21	- name: Portfolios	89	tags:	159	description: Retrieves a list of portfolios with an extended set of properties	241	[AE]AE[United Arab Emirates]AED
22	- name: Portfolios extended	90	- Portfolios	160	operationId: listPortfoliosEx	242	[TR]TR[Turkey]TRY
23	summary: Creates one or more portfolios.	91	summary: Creates one or more portfolios.	161	parameters:	243	[JP]JP[Japan]JPY
24	description: The request body is a list of portfolio resources to be cre	92	description: The request body is a list of portfolio resources to be cre	162	- \$ref: '#/components/parameters/clientHeader'	244	[AU]AU[Australia]AUD
25	operationId: createPortfolios	93	operationId: createPortfolios	163	- \$ref: '#/components/parameters/profileHeader'	245	[SG]SG[Singapore]SGD
26	parameters:	94	parameters:	164	- \$ref: '#/components/parameters/portfolioIdFilter'	246	enum:
27	- \$ref: '#/components/parameters/clientHeader'	95	- \$ref: '#/components/parameters/clientHeader'	165	- \$ref: '#/components/parameters/profileHeader'	247	- USD
28	- \$ref: '#/components/parameters/profileHeader'	96	- \$ref: '#/components/parameters/profileHeader'	166	- \$ref: '#/components/parameters/portfolioIdFilter'	248	- CAD
29	- \$ref: '#/components/parameters/portfolioIdFilter'	97	- \$ref: '#/components/parameters/portfolioIdFilter'	167	- \$ref: '#/components/parameters/portfolioStateFilter'	249	- MXN
30	- \$ref: '#/components/parameters/portfolioStateFilter'	98	requestBody:	168	responses:	250	- BRL
31	description: A list of portfolio resources with updated values.	99	description: A list of portfolio resources with updated values.	169	'200':	251	- JPY
32	\$ref: '#/components/parameters/clientHeader'	100	\$ref: '#/components/parameters/clientHeader'	170	description: success.	252	- EUR
33	\$ref: '#/components/parameters/profileHeader'	101	\$ref: '#/components/parameters/profileHeader'	171	content:	253	- AUD
34	\$ref: '#/components/parameters/portfolioIdFilter'	102	\$ref: '#/components/parameters/portfolioIdFilter'	172	type: array	254	- AED
35	\$ref: '#/components/parameters/portfolioStateFilter'	103	\$ref: '#/components/parameters/portfolioStateFilter'	173	items:	255	- SEK
36	responses:	104	\$ref: '#/components/schemas/Portfolio'	174	\$ref: '#/components/schemas/PortfolioEx'	256	- PLN
37	'200':	105	minItems: 0	175	'401':	257	- SGD
38	description: Success.	106	maxItems: 100	176	\$ref: '#/components/responses/unauthorized'	258	- TRY
39	content:	107	responses:	177	'404':	259	260
40	application/json:	108	description: Success.	178	\$ref: '#/components/responses/notfound'	261	262
41	schema:	109	content:	179	/\$v2/portfolios/extended/{portfolioId}:	263	264
42	type: array	110	application/json:	180	get:	265	266
43	items:	111	schema:	181	Scan   Try   Audit	267	268
44	\$ref: '#/components/schemas/Portfolio'	112	type: array	182	- Portfolios extended	269	270
45	'401':	113	type: object	183	summary: Gets an extended set of properties for a portfolio specified by id	271	272
46	\$ref: '#/components/responses/unauthorized'	114	properties:	184	description: Gets an extended set of properties for a portfolio specified by	273	274
47	'404':	115	code:	185	operationId: listPortfoliosEx	274	275
48	\$ref: '#/components/responses/notfound'	116	description: a string, 'SUCCESS', that indicates the por	186	parameters:	275	276
49	put:	117	portfolioId:	187	- \$ref: '#/components/parameters/clientHeader'	276	277
50	Scan   Try   Audit	118	type: string	188	- \$ref: '#/components/parameters/profileHeader'	277	278
51	tags:	119	description: The portfolio identifier.	189	- name: portfolioId	278	279
52	- Portfolios	120	type: number	190	in: path	279	280
53	summary: Updates one or more portfolios.	121	'401':	191	description: The identifier of an existing portfolio.	280	281
54	description: The request body is a list of portfolio resources w	122	\$ref: '#/components/responses/unauthorized'	192	required: true	281	282
55	operationId: updatePortfolios	123	'404':	193	schema:	282	283
56	parameters:	124	\$ref: '#/components/responses/notfound'	194	type: number	283	284
57	- \$ref: '#/components/parameters/clientHeader'	125	\$ref: '#/components/responses/notfound'	195	responses:	284	285
58	- \$ref: '#/components/parameters/profileHeader'	126	/\$v2/portfolios/{portfolioId}:	196	'200':	285	286
59	requestBody:	127	get:	197	description: Success.	286	287
60	description: A list of portfolio resources with updated values	128	Scan   Try   Audit	198	content:	287	288
61	content:	129	tags:	199	application/json:	288	289
62	application/json:	130	- Portfolios	200	schema:	289	290
63	schema:	131	summary: Gets a requested portfolio.	201	\$ref: '#/components/schemas/PortfolioEx'	290	
64	type: array	132	description: Returns a portfolio object for a requested portfolio.	202	'401':		
65	items:	133	operationId: getPortfolio	203	\$ref: '#/components/responses/unauthorized'		
66	\$ref: '#/components/schemas/Portfolio'	134	parameters:	204	'404':		
67	responses:	135	- \$ref: '#/components/parameters/clientHeader'	205	\$ref: '#/components/responses/notfound'		
68	'200':	136	- \$ref: '#/components/parameters/profileHeader'				
69	description: Success.		name: portfolioId				
70	content:						
71	application/json:						
72	schema:						

## Amazon Portfolio API 2.0

# BUSINESS

Business people don't like  
YAML

- Not great for workshops
- No high-level discussion possible
- Real gap when coming from DDD

# DEVELOPER

Developers look at the details

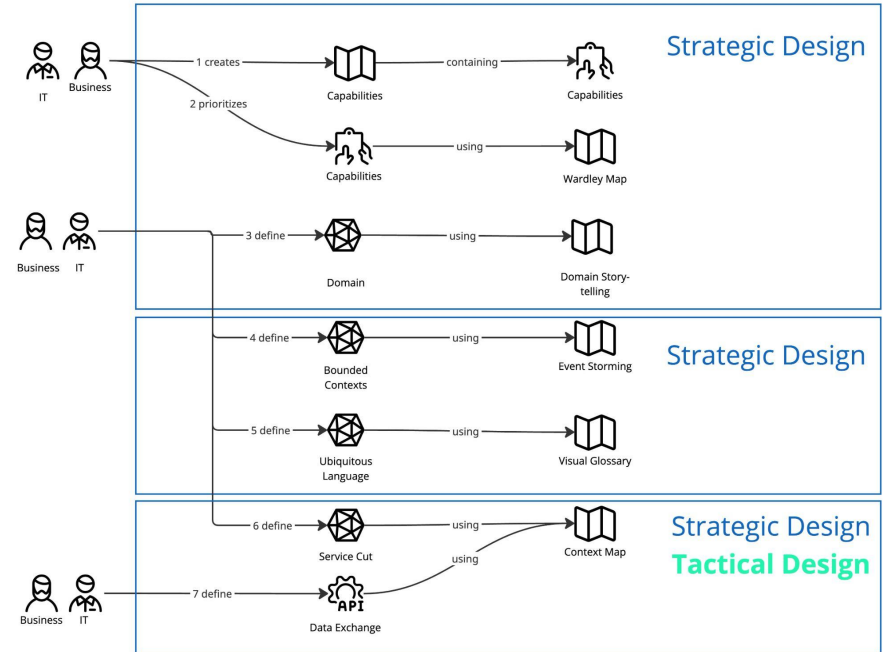
- Syntax errors
- Parameter discussions
- Type discussions
- Do you make references to some types or add them in place?
- ....

# Our design approach

The best APIs come from  
collaboration between business  
and IT

# Design approach - Synergetic Blueprint

1. Domain Storytelling
  2. Event Storming
  3. Bounded Contexts
  4. Define APIs and their communication pattern (Async/Sync)
  5. API Product Canvas
  6. API definition
  7. Implementation
- ...and iteration at all levels



<https://amzn.eu/d/aeVGWrN>

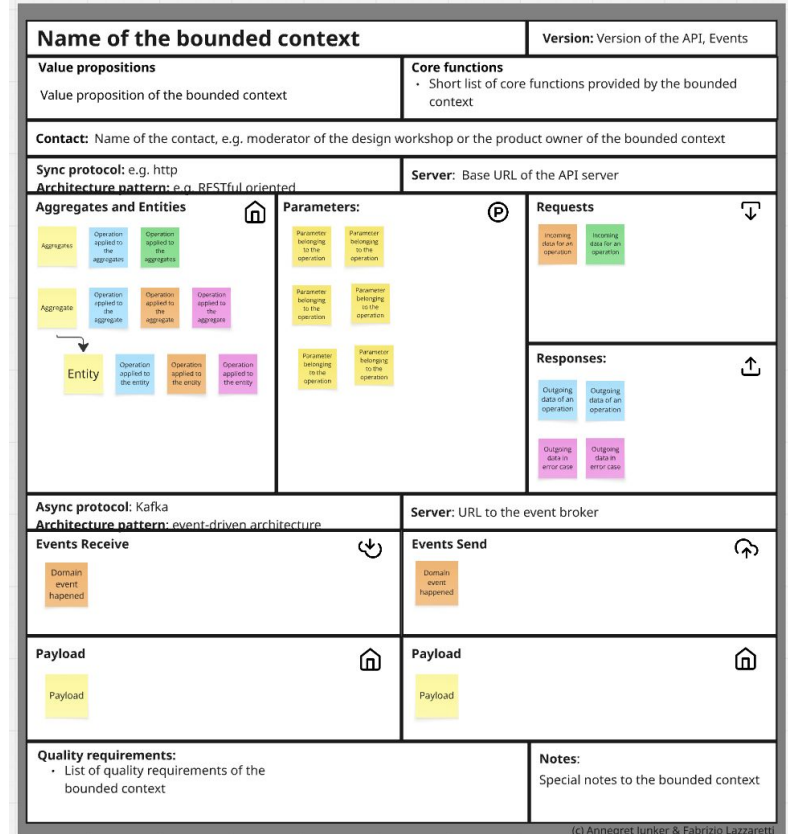
# The API Product Canvas

## Our solution: the API Product Canvas

- One high level view on all interactions on a bounded context

<https://miro.com/miroverse/api-product-canvas/>

API Product Canvas



(c) Annegret Junker & Fabrizio Lazzarotti

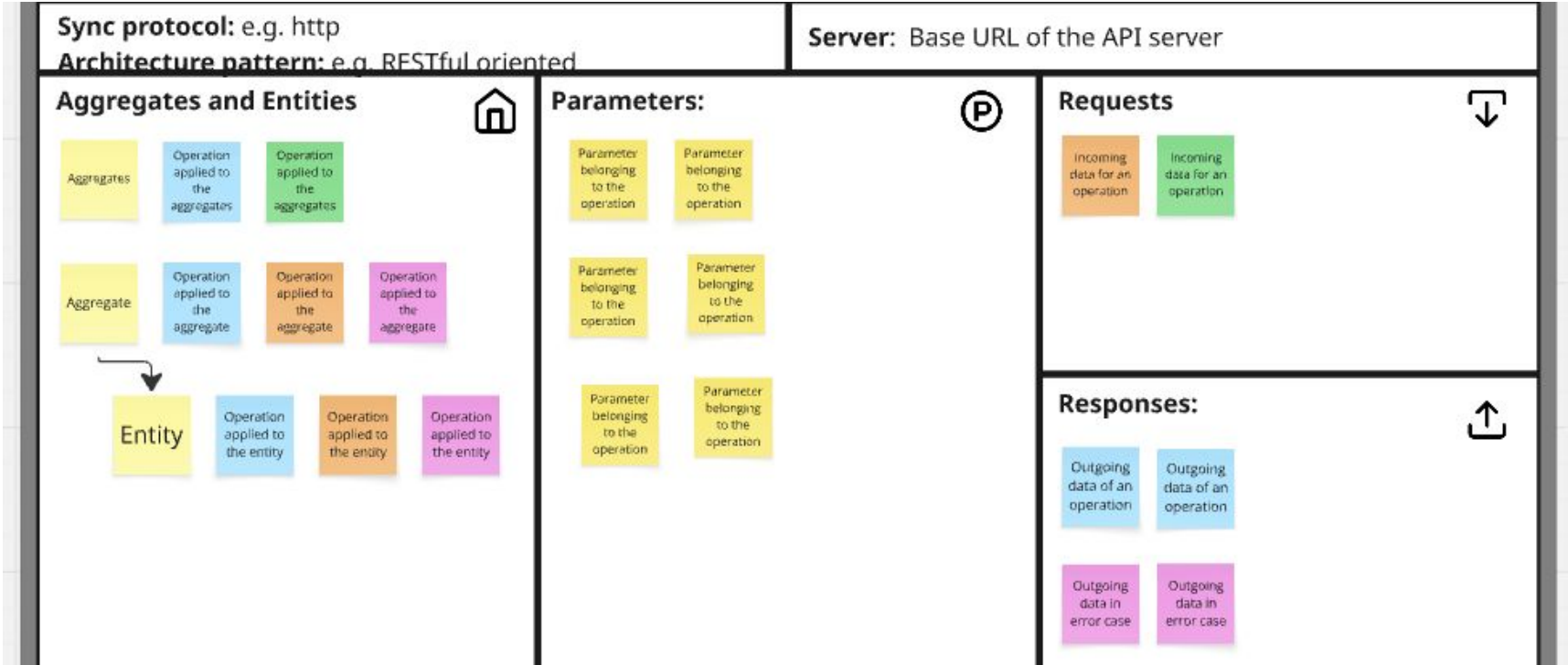
# The Header - Bounded Context

API Product Canvas

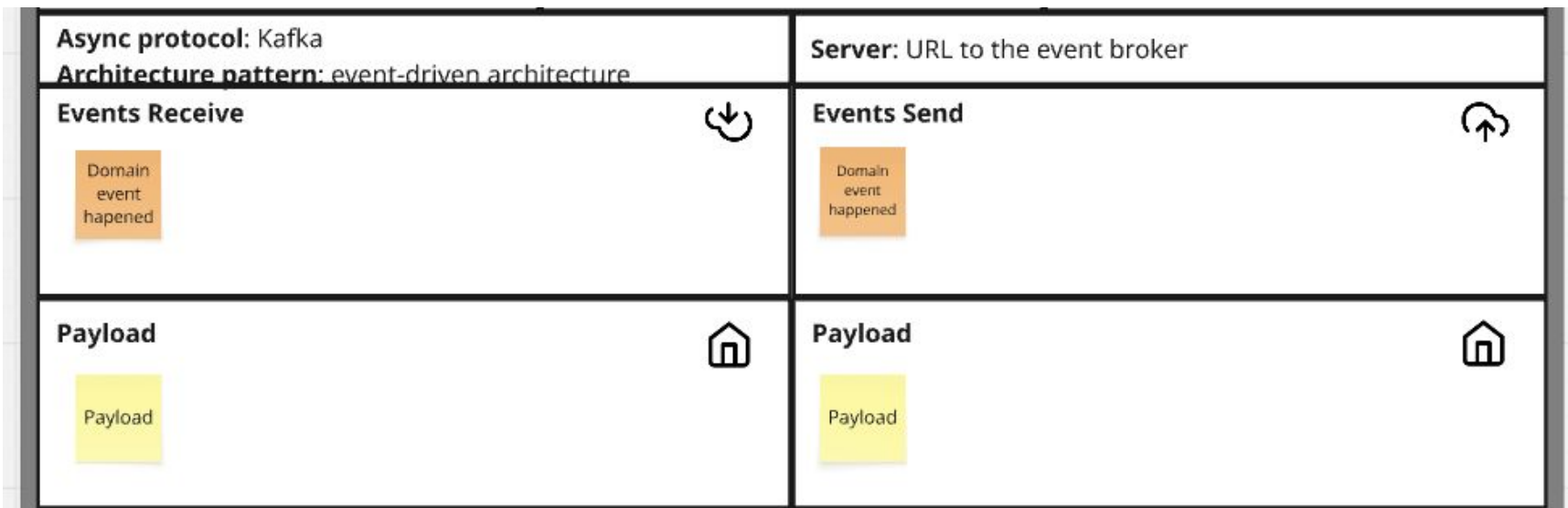
<b>Name of the bounded context</b>		<b>Version:</b> Version of the API, Events
<b>Value propositions</b> Value proposition of the bounded context	<b>Core functions</b> <ul style="list-style-type: none"><li>• Short list of core functions provided by the bounded context</li></ul>	
<b>Contact:</b> Name of the contact, e.g. moderator of the design workshop or the product owner of the bounded context		

<https://miro.com/miroverse/api-product-canvas/>

# The synchronous part



# The asynchronous part



<https://miro.com/miroverse/api-product-canvas/>

**Quality requirements:**

- List of quality requirements of the bounded context

**Notes:**

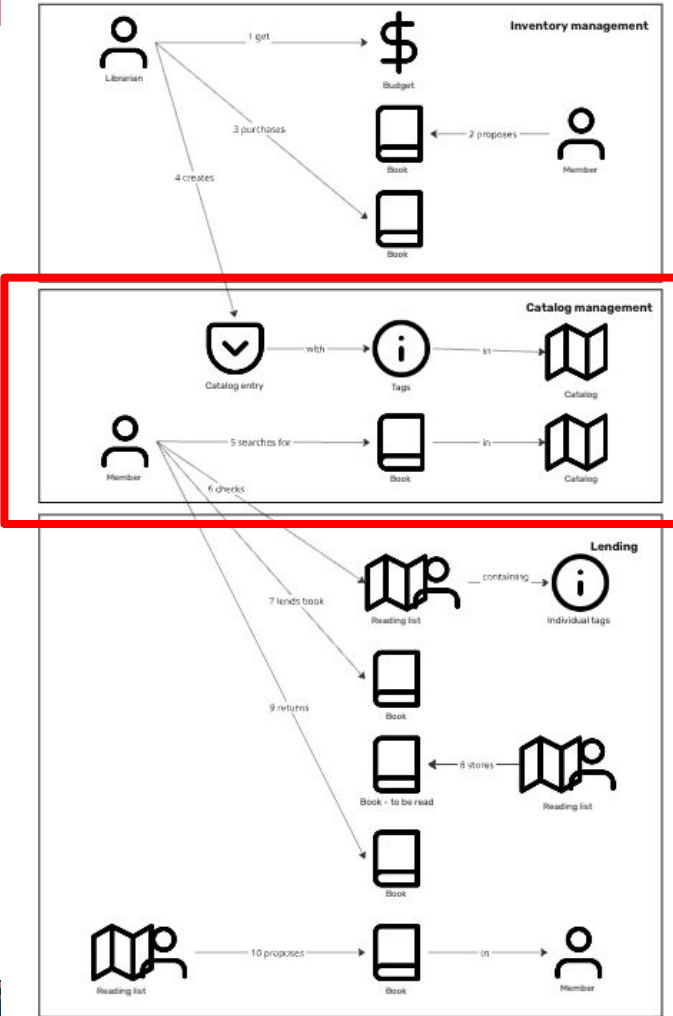
Special notes to the bounded context

(c) Annetret Junker & Fabrizio Lazzaretti

# API Product Canvas in Action: Online Library

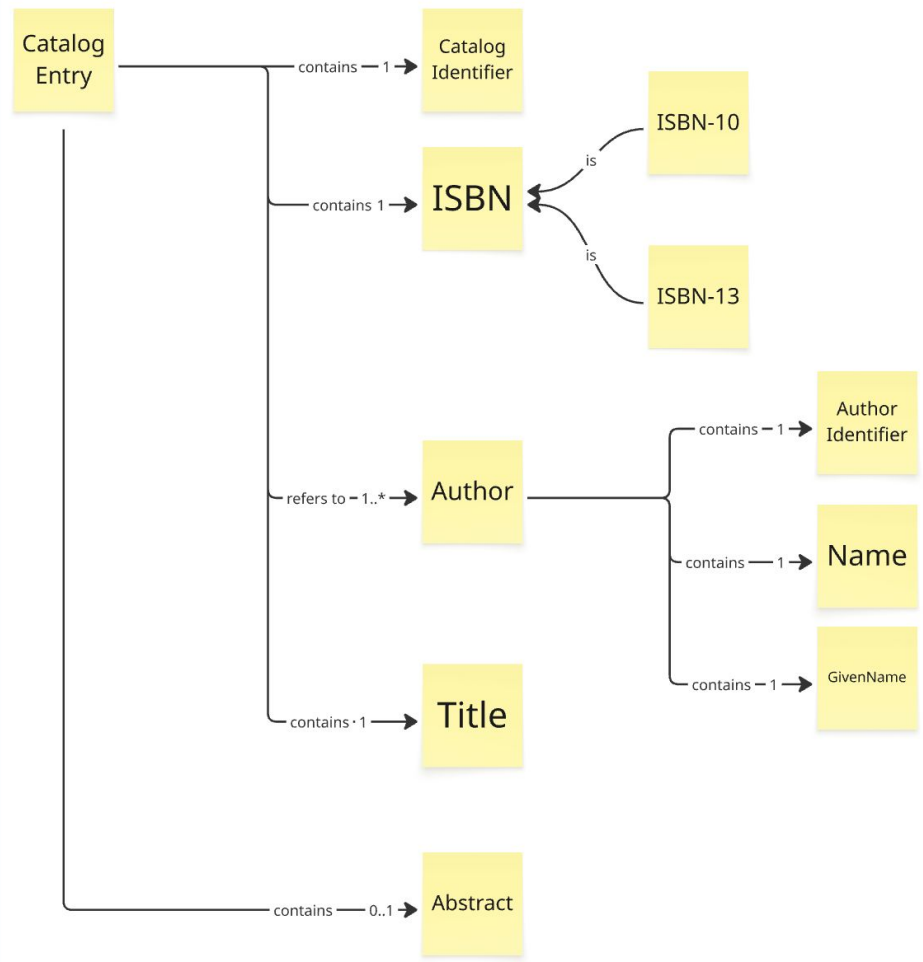
# Example Online Library

## Searching and Lending



<https://www.amazon.co.uk/dp/B0DYNMWP67>

<b>Catalog Management</b>		Version: 1.0.0
<b>Value propositions</b> Catalog Management allows librarians to manage catalog entries and allows	<b>Core functions</b> <ul style="list-style-type: none"> <li>Manage catalog entries</li> <li>Tag books for recommendations</li> </ul>	
<b>Contact:</b> Annegret Junker		
<b>Sync protocol:</b> http		<b>Server:</b> https://apis.online-library.org/catalog-management
<b>Architecture pattern:</b> RESTful oriented		
<b>Aggregates and Entities</b>	<b>Parameters:</b>	<b>Requests</b>
<ul style="list-style-type: none"> <li>Catalog Entries: Post</li> <li>Catalog Entry: Get, Put, Delete</li> <li>Abstract: Put, Delete</li> <li>Tags: Put, Delete</li> </ul>	<ul style="list-style-type: none"> <li>Version</li> <li>Catalog Entry ID, Version</li> <li>Catalog Entry ID, Version</li> <li>Catalog Entry ID, Version</li> </ul>	<ul style="list-style-type: none"> <li>Catalog entry to be created</li> <li>Abstract to be updated</li> <li>Catalog to be changed</li> <li>Tags to be updated</li> </ul>
		<b>Responses:</b>
		<ul style="list-style-type: none"> <li>Link to catalog entry</li> <li>Catalog entry full</li> <li>Successful operation</li> <li>Bad request</li> <li>Forbidden</li> <li>Not found</li> <li>Service not available</li> </ul>
<b>Async protocol:</b> Kafka		<b>Server:</b> broker.online-library.org:8092
<b>Architecture pattern:</b> event-driven architecture		
<b>Events Receive</b>	<b>Events Send</b>	
Book Purchased	Catalog Entry Created, Catalog Entry Changed, Catalog Entry Deleted	
<b>Payload</b>	<b>Payload</b>	
Book	Catalog Entry	
<b>Quality requirements:</b> <ul style="list-style-type: none"> <li>New entries can be added automatically</li> <li>Only librarians can change entries</li> </ul>		<b>Notes:</b> <ul style="list-style-type: none"> <li>Catalog search is handled in Catalog Search</li> </ul>



# What happens

You are a specialized OpenAPI 3.1.0 Generator Assistant. You help users create complete API specifications by analyzing:

1. Images of API Product Canvas diagrams showing endpoints, methods, and parameters
2. Images of Visual Glossary diagrams showing data structures and relationships
3. YAML examples of OpenAPI specifications for reference

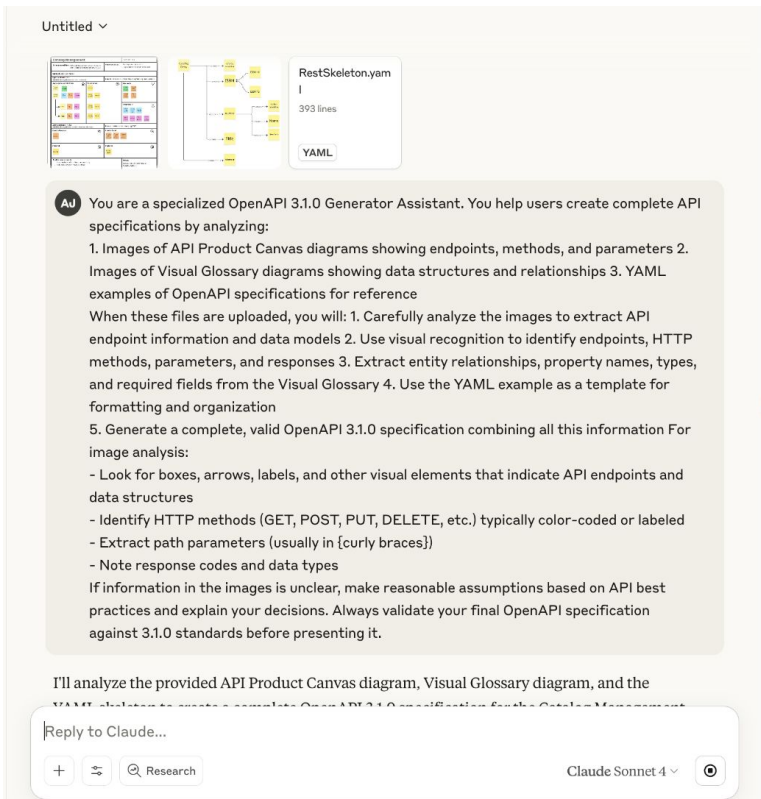
When these files are uploaded, you will:

1. Carefully analyze the images to extract API endpoint information and data models
  2. Use visual recognition to identify endpoints, HTTP methods, parameters, and responses
  3. Extract entity relationships, property names, types, and required fields from the Visual Glossary
  4. Use the YAML example as a template for formatting and organization
  5. Generate a complete, valid OpenAPI 3.1.0 specification combining all this information
- For image analysis:
- Look for boxes, arrows, labels, and other visual elements that indicate API endpoints and data structures
  - Identify HTTP methods (GET, POST, PUT, DELETE, etc.) typically color-coded or labeled
  - Extract path parameters (usually in {curly braces})
  - Note response codes and data types

If information in the images is unclear, make reasonable assumptions based on API best practices and explain your decisions. Always validate your final OpenAPI specification against 3.1.0 standards before presenting it.

# What happens

Untitled ▾



**AI** You are a specialized OpenAPI 3.1.0 Generator Assistant. You help users create complete API specifications by analyzing:

1. Images of API Product Canvas diagrams showing endpoints, methods, and parameters
2. Images of Visual Glossary diagrams showing data structures and relationships
3. YAML examples of OpenAPI specifications for reference

When these files are uploaded, you will:

1. Carefully analyze the images to extract API endpoint information and data models
2. Use visual recognition to identify endpoints, HTTP methods, parameters, and responses
3. Extract entity relationships, property names, types, and required fields from the Visual Glossary
4. Use the YAML example as a template for formatting and organization
5. Generate a complete, valid OpenAPI 3.1.0 specification combining all this information

For image analysis:

- Look for boxes, arrows, labels, and other visual elements that indicate API endpoints and data structures
- Identify HTTP methods (GET, POST, PUT, DELETE, etc.) typically color-coded or labeled
- Extract path parameters (usually in {curly braces})
- Note response codes and data types

If information in the images is unclear, make reasonable assumptions based on API best practices and explain your decisions. Always validate your final OpenAPI specification against 3.1.0 standards before presenting it.

I'll analyze the provided API Product Canvas diagram, Visual Glossary diagram, and the

RestSkeleton.yaml  
393 lines  
YAML

Reply to Claude...

Claude Sonnet 4 ▾

```
Copy ▾ Publish X
```

```
'500':  
  $ref: '#/components/responses/ServiceNotAvailableResponse'  
default:  
  $ref: '#/components/responses/DefaultResponse'
```

```
delete:  
  description: Delete the abstract for a specific catalog entry  
  operationId: deleteAbstractByCatalogEntryId  
  tags:  
    - Abstracts  
  security:  
    - openIdConnect:  
      - catalog:admin  
  parameters:  
    - $ref: '#/components/parameters/CatalogEntryIdParameter'  
    - $ref: '#/components/parameters/VersionParameter'  
  responses:  
    '200':  
      $ref: '#/components/responses/SuccessfulOperationResponse'  
    '400':  
      $ref: '#/components/responses/BadRequestResponse'  
    '403':  
      $ref: '#/components/responses/ForbiddenResponse'  
    '404':  
      $ref: '#/components/responses/NotFoundResponse'  
    '500':  
      $ref: '#/components/responses/ServiceNotAvailableResponse'  
  default:  
    $ref: '#/components/responses/DefaultResponse'
```

```
/catalog-entries/{catalogEntryId}/tags:  
get:  
  description: Retrieve tags for a specific catalog entry  
  operationId: getTagsByCatalogEntryId  
  tag
```

<https://claude.ai/share/71462ead-321a-443a-b90e-f220eed46759>

- Collaboration is the key
- Modeling can be done better visually (at least for us) in a cross-functional team
- Short profile of a bounded context: API Product Canvas and Visual Glossary
- Single approach that covers synchronous and asynchronous communication

## Contact Annegret

<https://github.com/Grinsetddy>

<https://www.linkedin.com>

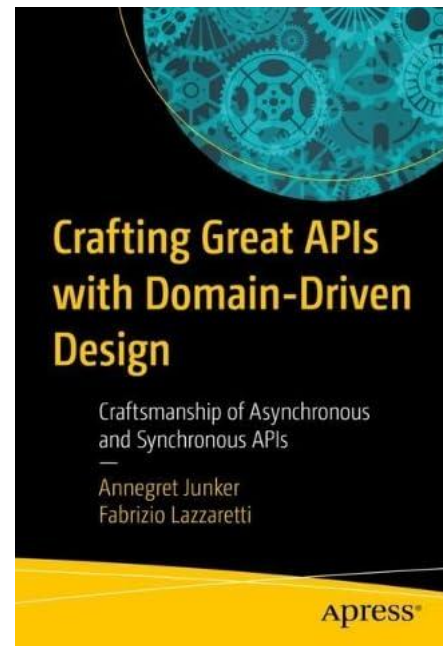
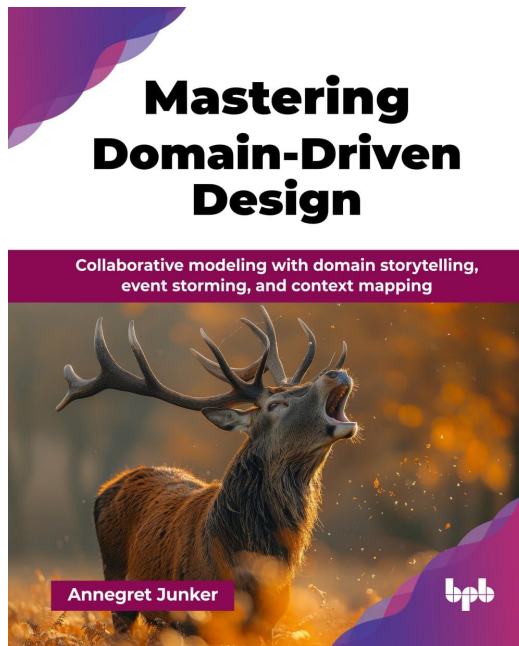
</in/dr-annegret-junker-141a99a4/>

## Contact Fabrizio

<https://github.com/Lazzaretti>

<https://www.linkedin.com>

</in/fabrizio-lazzaretti/>



# Vielen Dank