

Data Mesh

Datenanalysen selber machen



JOCHEN CHRIST
@JOCHEN_CHRIST

Hi,
I am Jochen

Jochen Christ

Software Engineer at INNOQ



Java

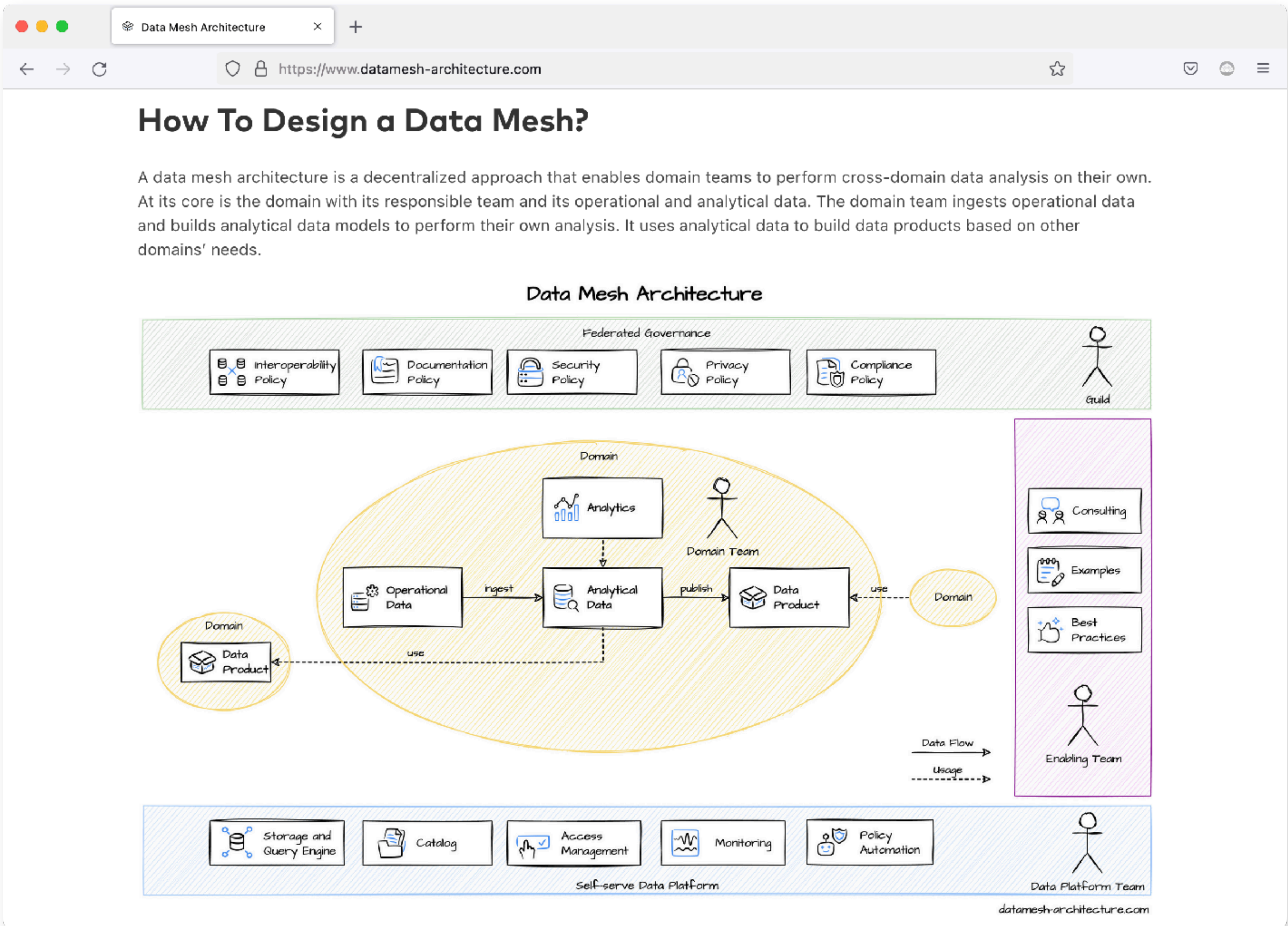


Self-contained Systems



Remote Mob Programming





O'REILLY®

Deutsche
Ausgabe

Data Mesh

Dezentrale Architekturen für die
Datenanalyse einrichten



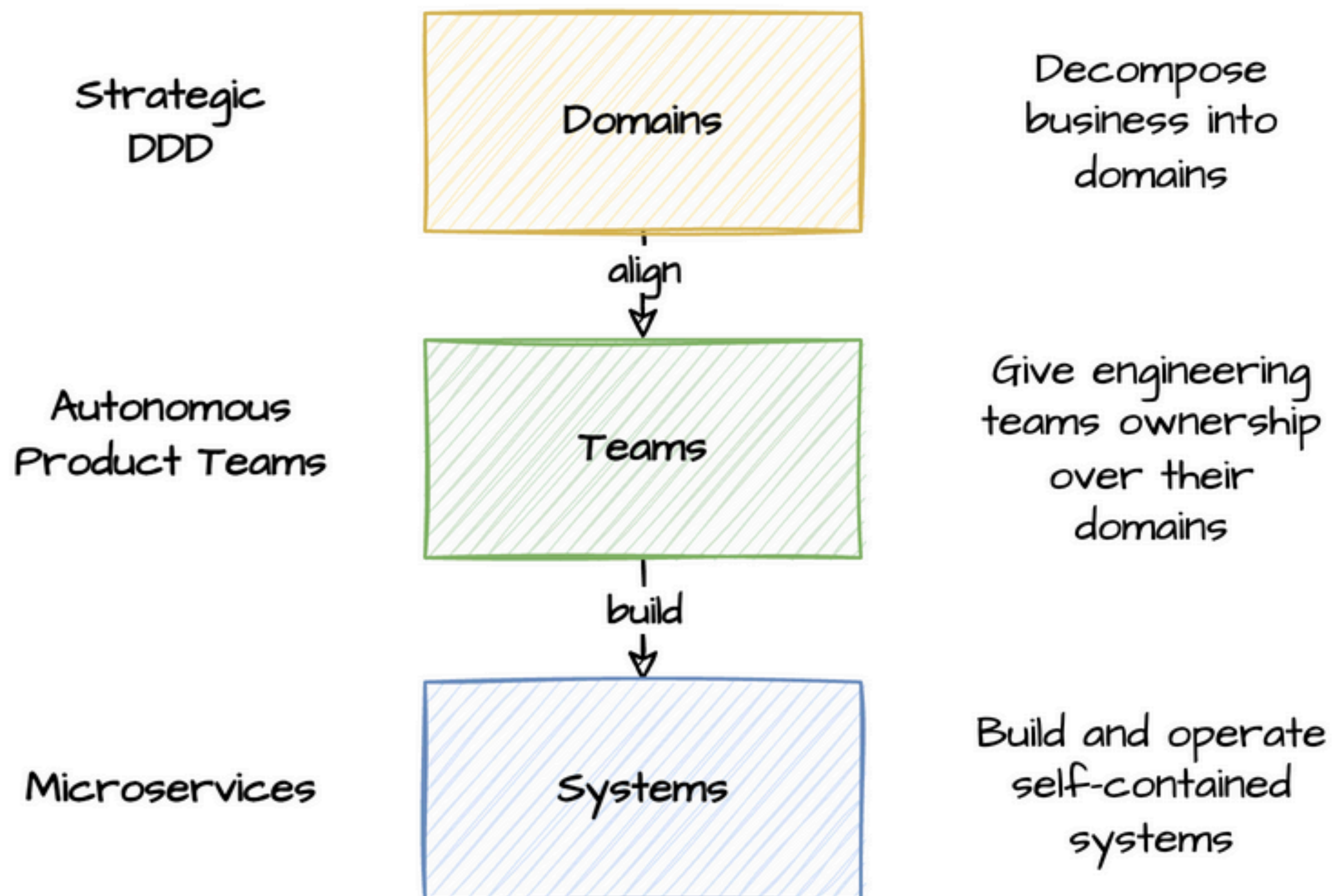
Zhamak Dehghani

Vorwort von Martin Fowler

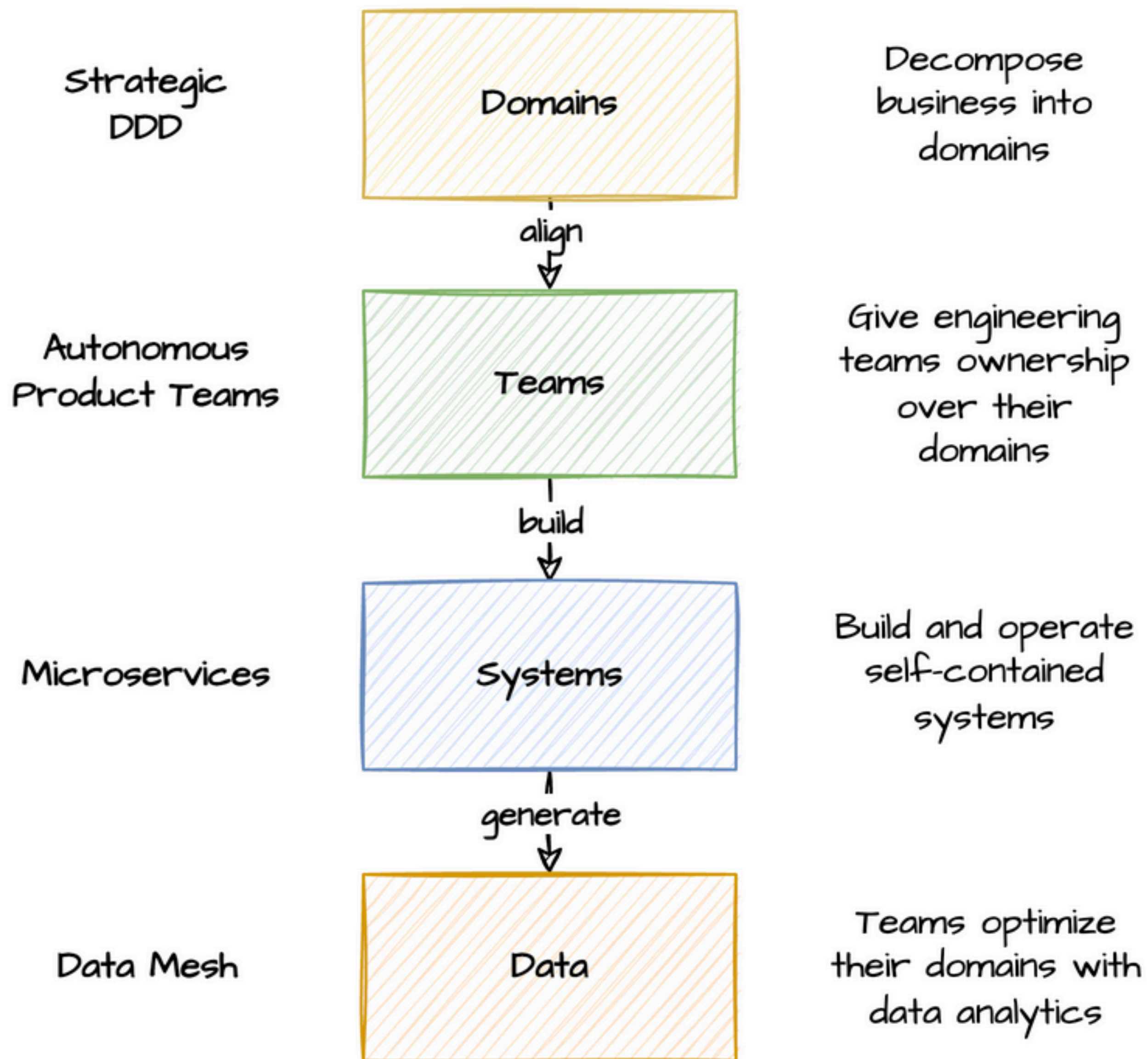
Übersetzung von Jochen Christ und Simon Harrer

**CENTRAL
DATA TEAMS
FAILED TO SCALE**

Decentralize to scale



Decentralize to scale



DATA MESH
DECENTRALIZES DATA,
SO YOU CAN DO
DATA ANALYSIS ON YOUR OWN

What Is Data Mesh?

Strategic
Domain-driven
Design

Socio-technical
Perspective

Technology

Domain
Ownership

Domain
Bounded Context

Domain Teams

Operational &
Analytical Data

Data as a
Product

Product Thinking

Data Product by
Domain Team

Interoperability
Interfaces

Self-serve
Data Platform

Domain-agnostic

Data Platform
Team

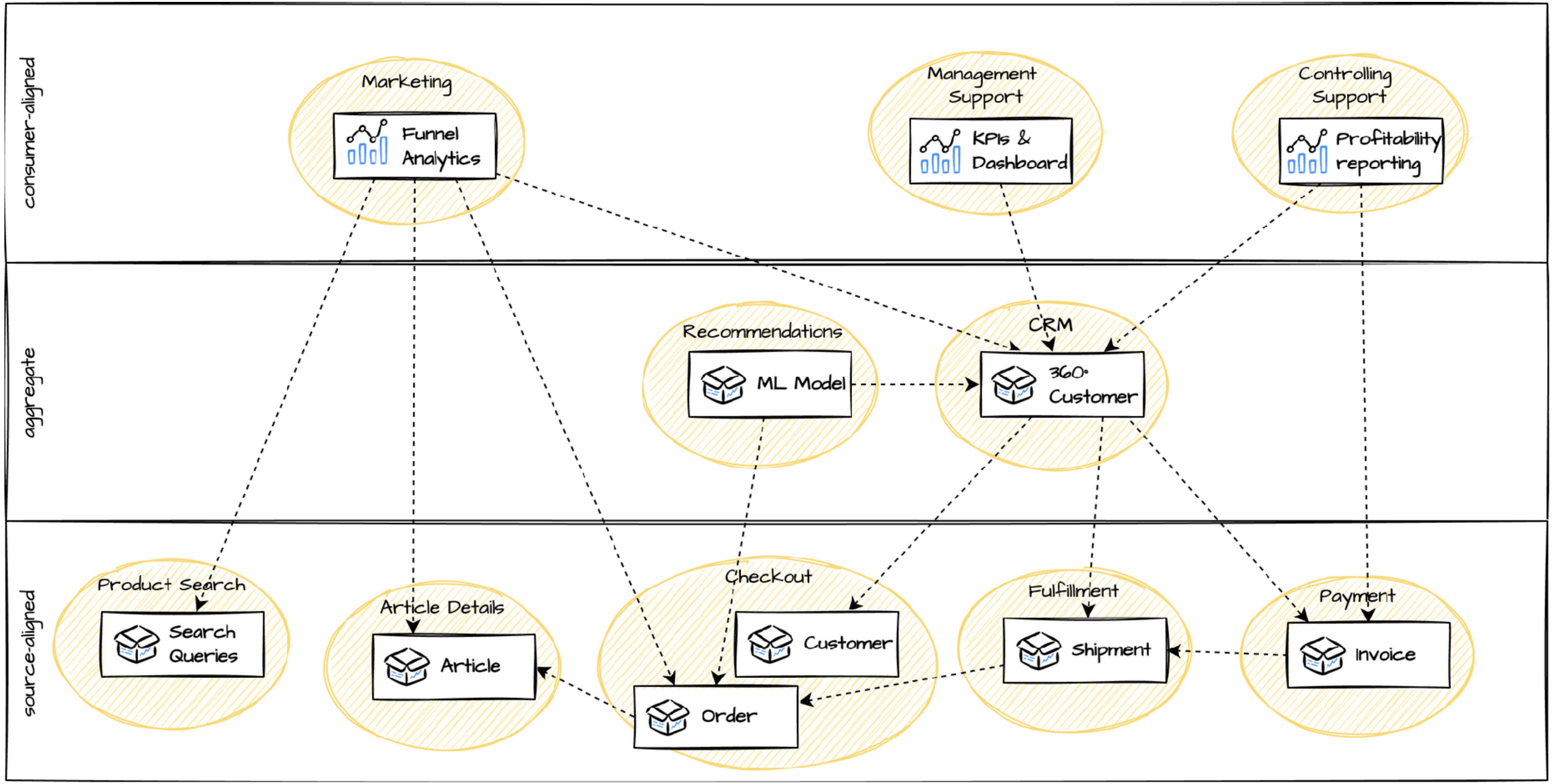
Self-serve
Data Platform

Federated
Governance

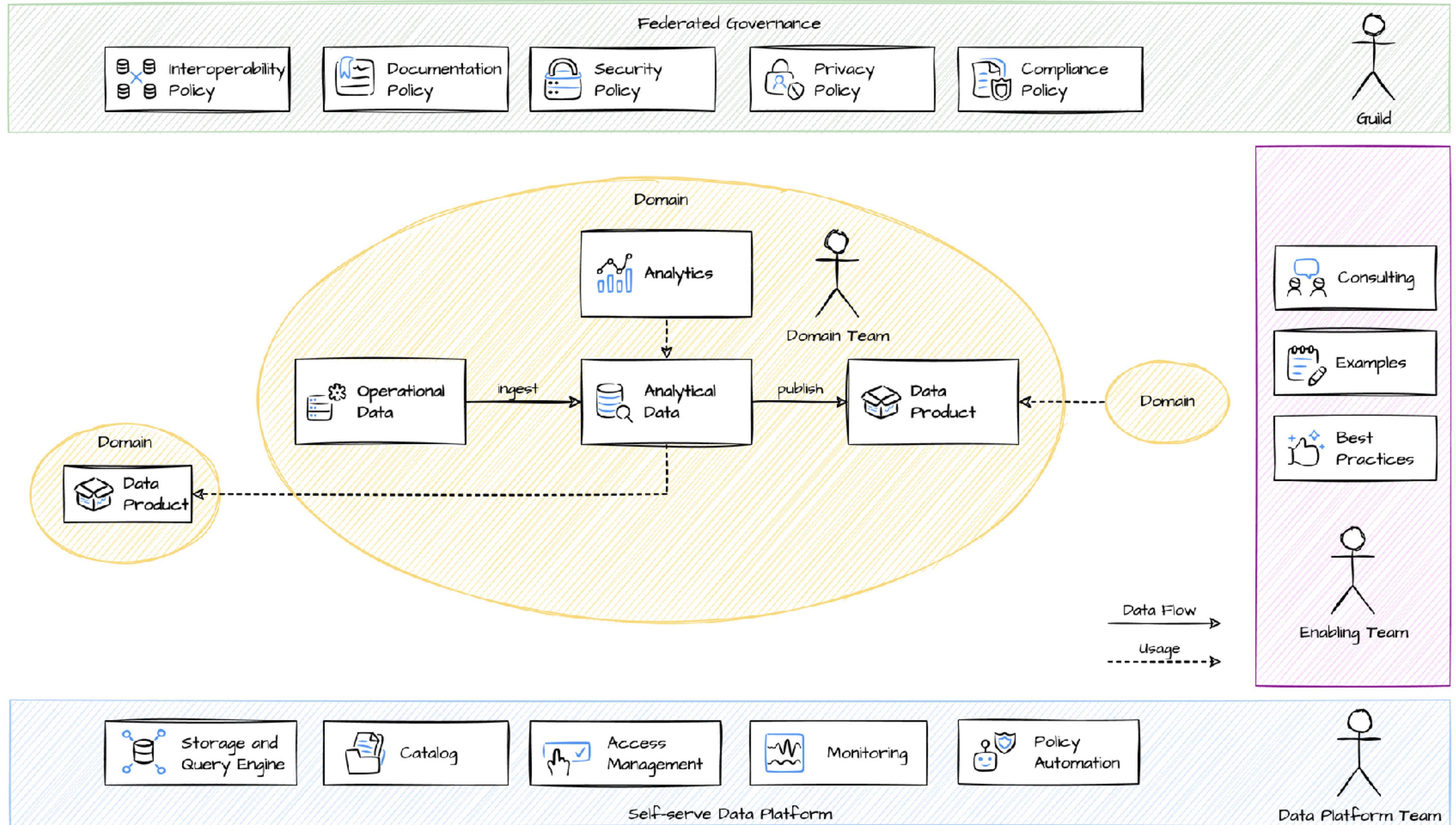
Context Mapping

Guild

Data Governance



Data Mesh Architecture



Data Product

Data

Row	sku	location	available	updated_at
1	9520010951145	20	0	2021-02-28 12:29:21 UTC
2	9520010951145	20	1	2021-03-02 09:07:21 UTC
3	9520010951145	20	0	2021-03-03 16:36:21 UTC
4	9520010951145	20	1	2021-03-04 13:03:21 UTC
5	9520010951145	20	2	2021-03-05 17:26:21 UTC
6	9520010951145	20	3	2021-03-06 03:35:21 UTC
7	9520010951145	20	2	2021-03-06 17:25:21 UTC
8	9520010951145	20	1	2021-03-07 18:10:21 UTC

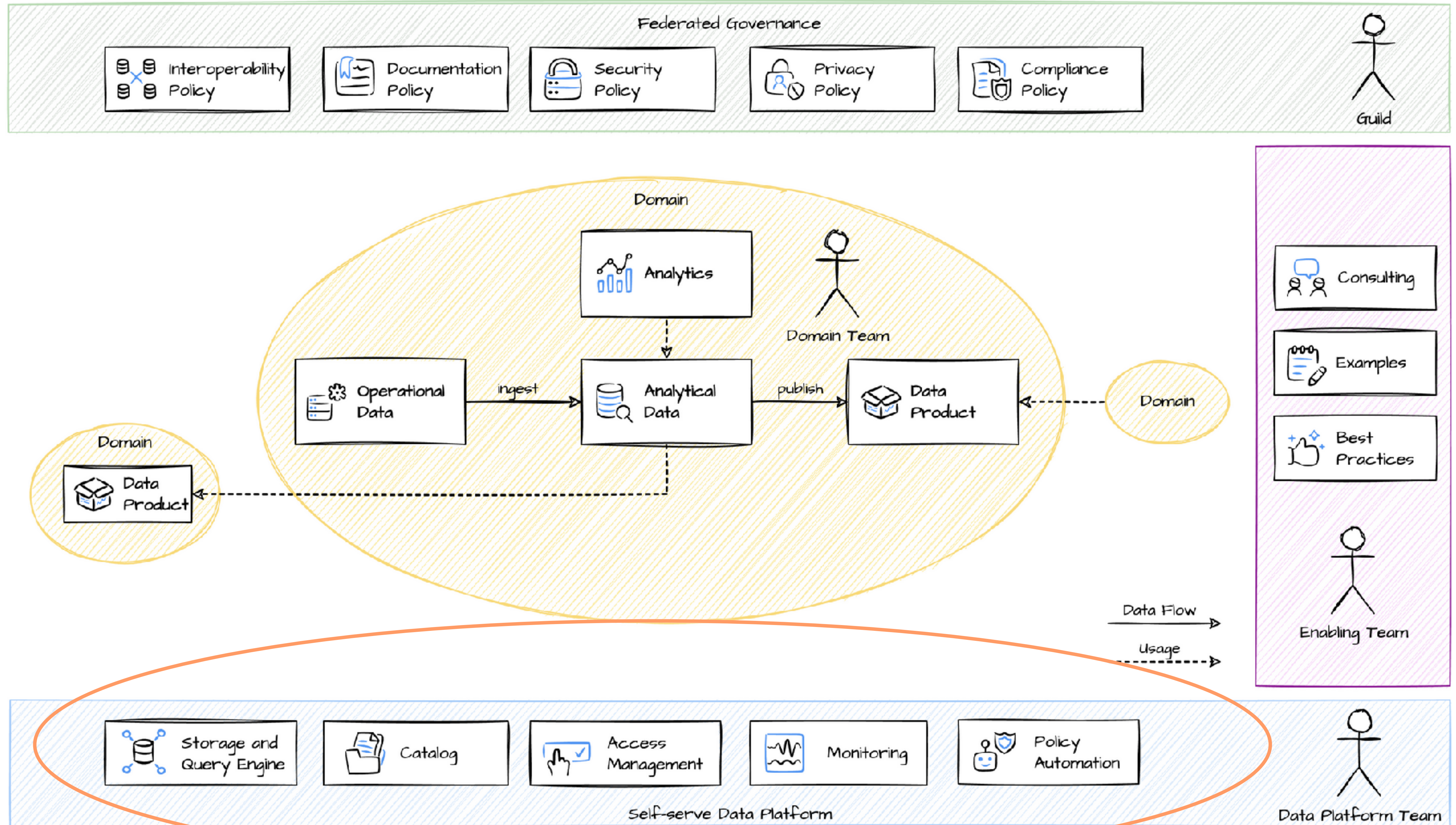
Metadata

Tags (1)	
Data Product	
Display name	Value
Domain Team	Fulfillment
Update Frequency	Hourly
Environment	PRODUCTION
Schema	
Filter Enter property name or value	
Field name	TypeMode
sku	STRINGNULLABLE
location	STRINGNULLABLE
available	INT64NULLABLE
updated_at	TIMESTAMPNULLABLE

Operations

132M	0	5 min ago	42	\$1.4k
Rows	Duplicates	Freshness	24h Usages	Monthly Costs

Data Mesh Architecture

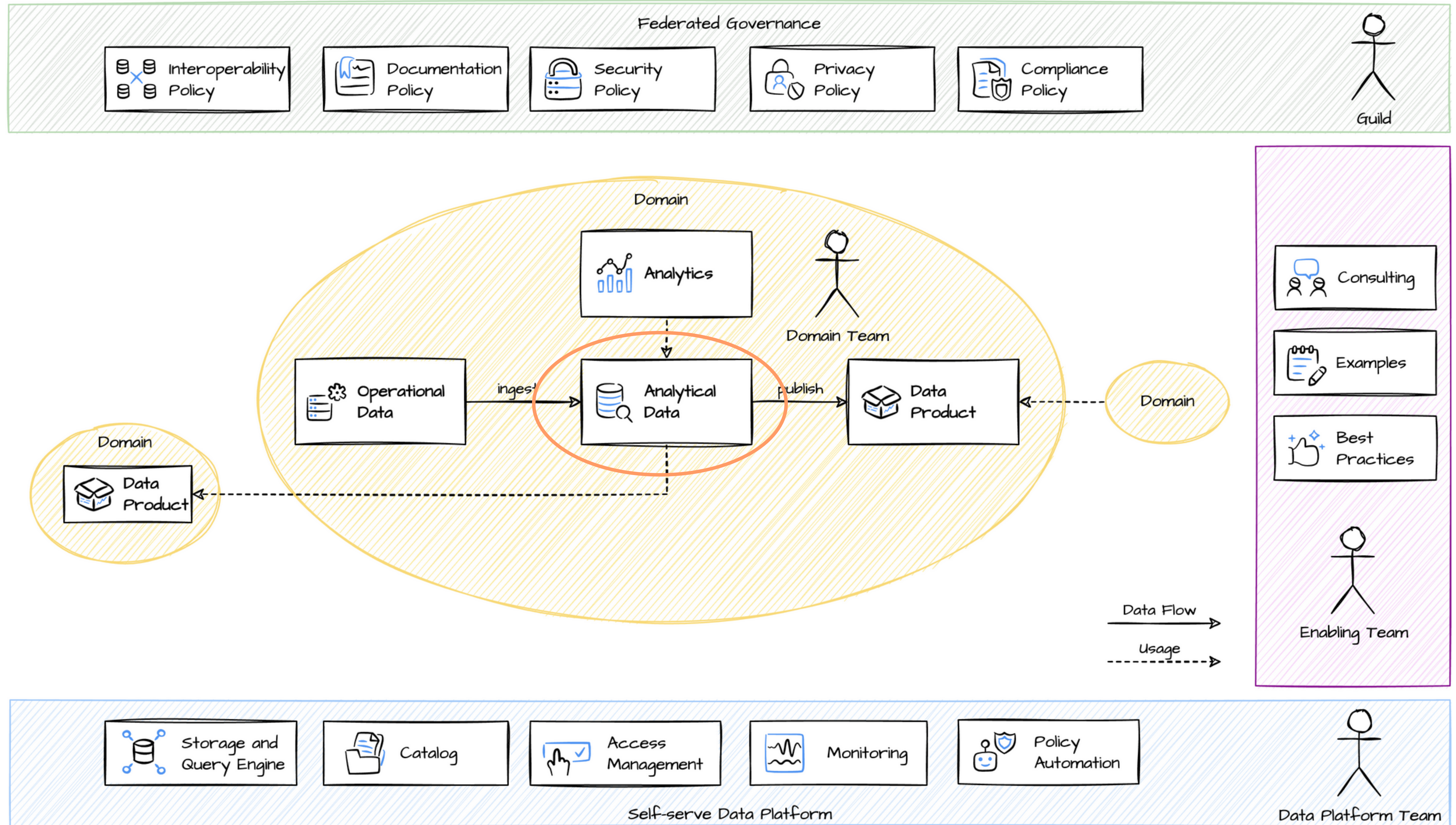




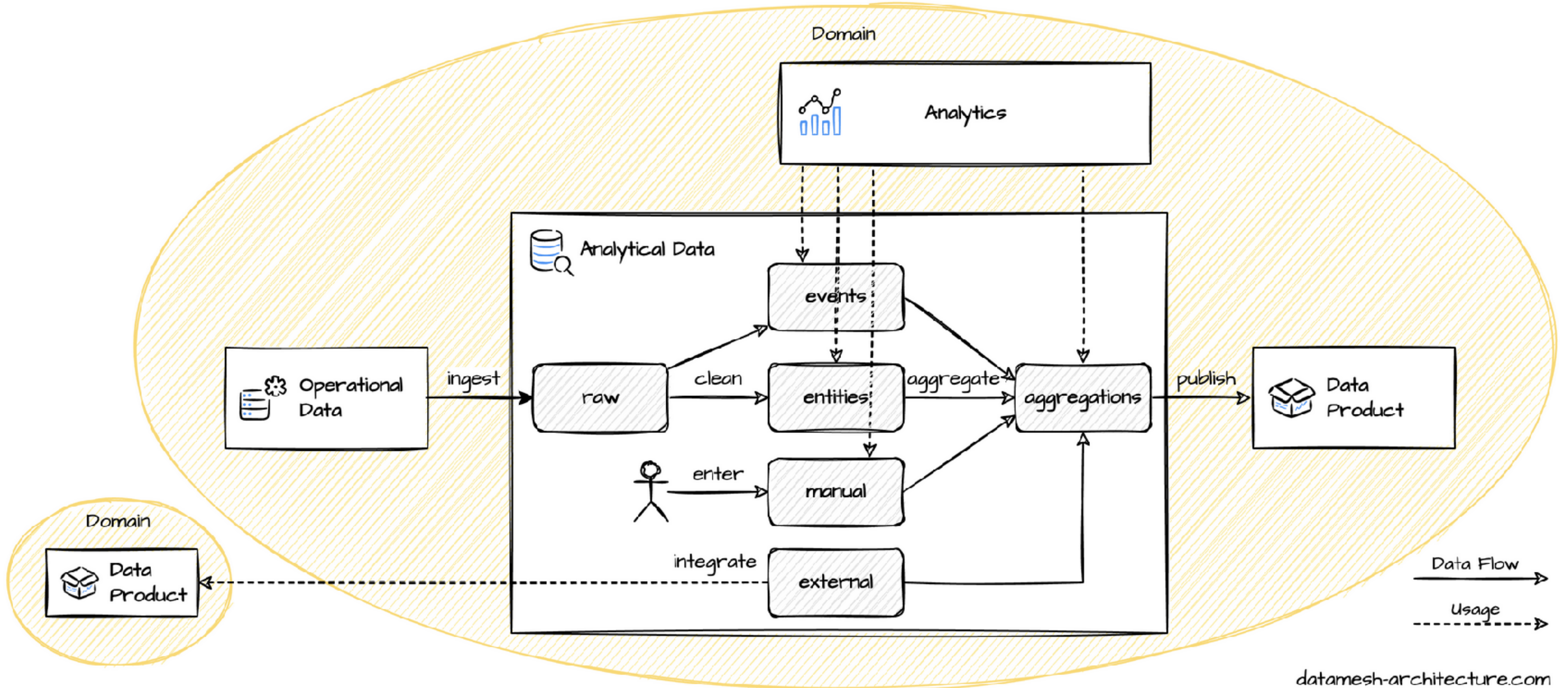
AWS Glue Data Catalog



Data Mesh Architecture



Analytical Data




```
1  -- Step 1: Deduplicate
2  WITH inventory_deduplicated AS (
3      SELECT *
4      EXCEPT (row_number)
5      FROM (
6          SELECT *,
7              ROW_NUMBER() OVER (PARTITION BY id ORDER BY time DESC) row_number
8          FROM `datameshexample-fulfillment.raw.inventory`)
9      WHERE row_number = 1
10 ),
11 -- Step 2: Parse JSON to columns
12 inventory_parsed AS (
13     SELECT
14         json_value(data, "$.sku") AS sku,
15         json_value(data, "$.location") AS location,
16         CAST(json_value(data, "$.available") AS int64) AS available,
17         CAST(json_value(data, "$.updated_at") AS timestamp) AS updated_at,
18     FROM inventory_deduplicated
19 )
20 -- Step 3: Actual Query
21 SELECT sku, location, available, updated_at
22 FROM inventory_parsed
23 ORDER BY sku, location, updated_at
```




dbtTM

5

EXPLORER

...

FULFILLMENT_DBT

> analyses

> dbt_packages

> logs

> macros

> models

> analytics

> data_products

> staging / inventory

! src_inventory.yml

stg_inventory__inven...

! stg_inventory.yml

> seeds

> snapshots

> target

> tests

inventory_history_test....

.gitignore

! dbt_project.yml

i README.md

> OUTLINE

> TIMELINE

stg_inventory__inventory_history.sql

inventory_history.sql

inventory_latest.sql

inventory__avai

...

models > staging > inventory > stg_inventory__inventory_history.sql

1 with source as (
2 | select * from {{ source('fulfillment_sources', 'inventory') }}
3 |),
4
5 parsed as (
6 | select
7 | record_metadata:"CreateTime"::number as metadata_createtime,
8 | record_metadata:"headers"."ce-id"::string as event_id,
9 | record_metadata:"headers"."ce-source"::string as event_source,
10 | record_metadata:"headers"."ce-subject"::string as event_subject,
11 | record_metadata:"headers"."ce-time"::timestamp as event_time,
12 | record_metadata:"headers"."ce-type"::string as event_type,
13 | record_metadata:"key"::string as kafka_key,
14 | record_metadata:"offset"::number as kafka_offset,
15 | record_metadata:"partition"::number as kafka_partition,
16 | record_metadata:"topic"::string as kafka_topic,
17 | record_content:available::number as available,
18 | record_content:location::string as location,

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

zsh + ▾ □ □ ✖ ^ ×

jochen@Jochens-MacBook-Pro-2 fulfillment_dbt % dbt run

0 0 AWS

Ln 31, Col 3 Spaces: 4 UTF-8 LF SQL

🗨 🔔

5

EXPLOLER

...

FULFILLME...

> analyses

> dbt_packages

> logs

> macros

> models

> seeds

> snapshots

> target

> tests

inventory_history_test....

.gitignore

dbt_project.yml

README.md

OUTLINE

TIMELINE

tests > inventory_history_test.sql

1

2

3

4

5

with inventory_history_entry as (
select count(*) c
from {{ref('inventory_history')}}
)
select * from inventory_history_entry where c < 2000000

PROBLEMS

OUTPUT

DEBUG CONSOLE

TERMINAL

zsh

+

^

×

14:41:14

8 of 11

PASS

not_null_stg_inventory__inventory_history_sku.....

[PASS in 0.45s]

14:41:14

9 of 11

START

test not_null_stg_inventory__inventory_history_updated_at.....

[RUN]

14:41:20

9 of 11

PASS

not_null_stg_inventory__inventory_history_updated_at.....

[PASS in 6.23s]

14:41:20

10 of 11

START

test source_not_null_fulfillment_sources_inventory_RECORD_CONTENT

[RUN]

14:41:21

10 of 11

PASS

source_not_null_fulfillment_sources_inventory_RECORD_CONTENT.....

[PASS in 0.92s]

14:41:21

11 of 11

START

test source_not_null_fulfillment_sources_inventory_RECORD_METADATA

[RUN]

14:41:23

11 of 11

PASS

source_not_null_fulfillment_sources_inventory_RECORD_METADATA.....

[PASS in 1.70s]

14:41:23

Finished running 11 tests in 15.67s.

14:41:23

Completed successfully

14:41:23

Done. PASS=11 WARN=0 ERROR=0 SKIP=0 TOTAL=11

jochen@Jochens-MacBook-Pro-2 fulfillment_dbt %

0

0

AWS

Ln 5, Col 11

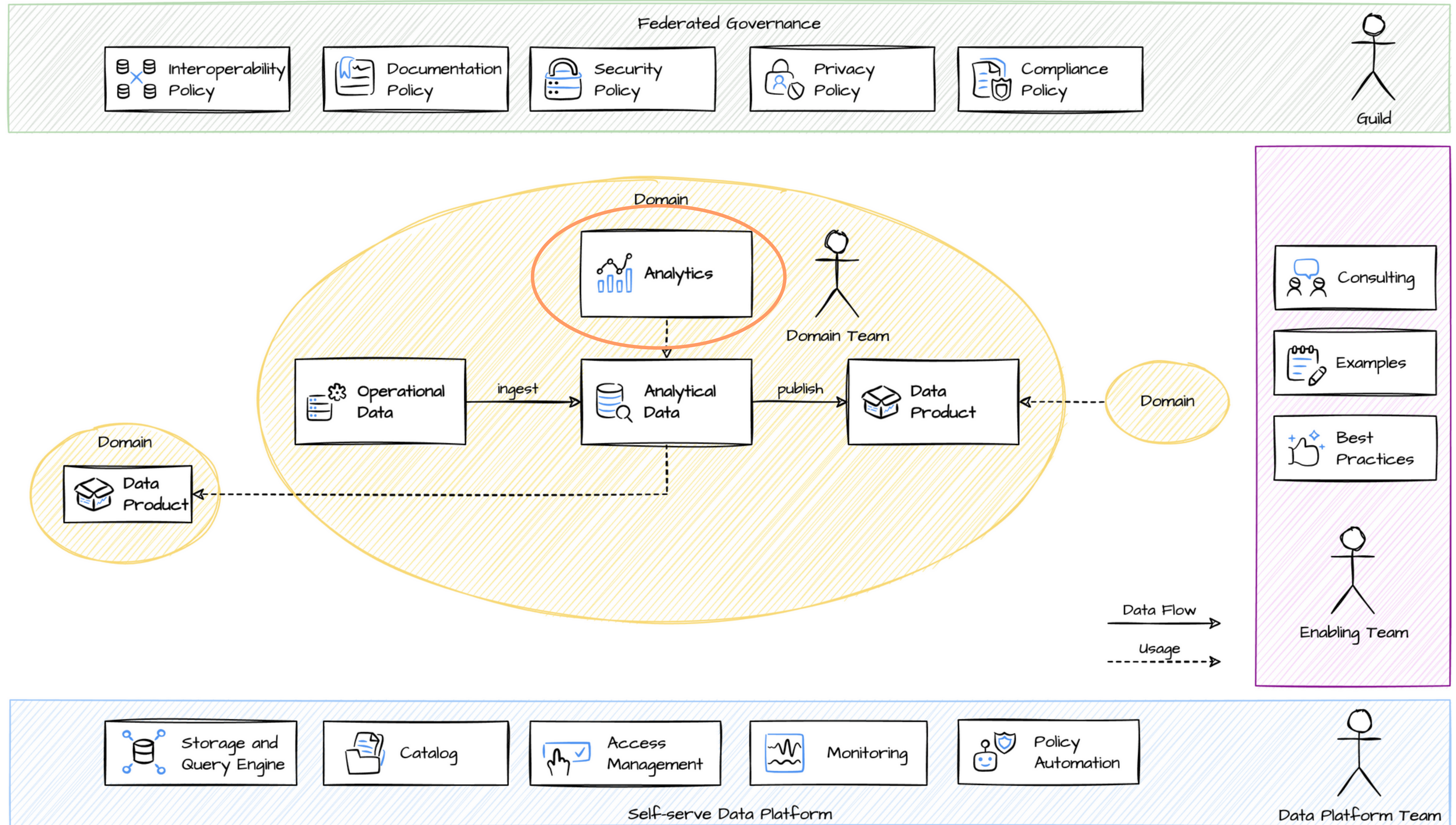
Spaces: 4

UTF-8

LF

SQL

Data Mesh Architecture





Visualize



Filter



Parameters



Actions



Themes



Settings



Community

Dataset

SPICE Business Review 100%

Fields list

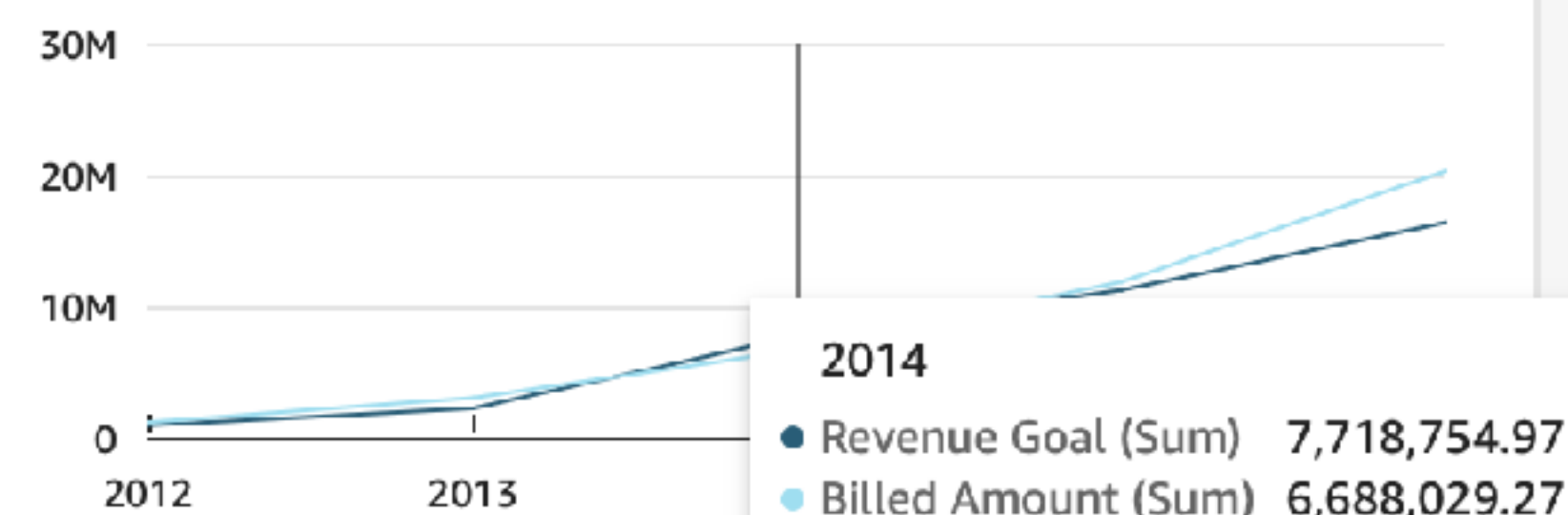
Search fields

- # Billed Amount
- Channel
- # Cost
- Customer ID
- Customer Name
- Customer Region
- Date
- # Distinct ID
- # Revenue Goal
- Segment
- Service Line

Field wells

Sheet 1 +

Revenues vs Goals

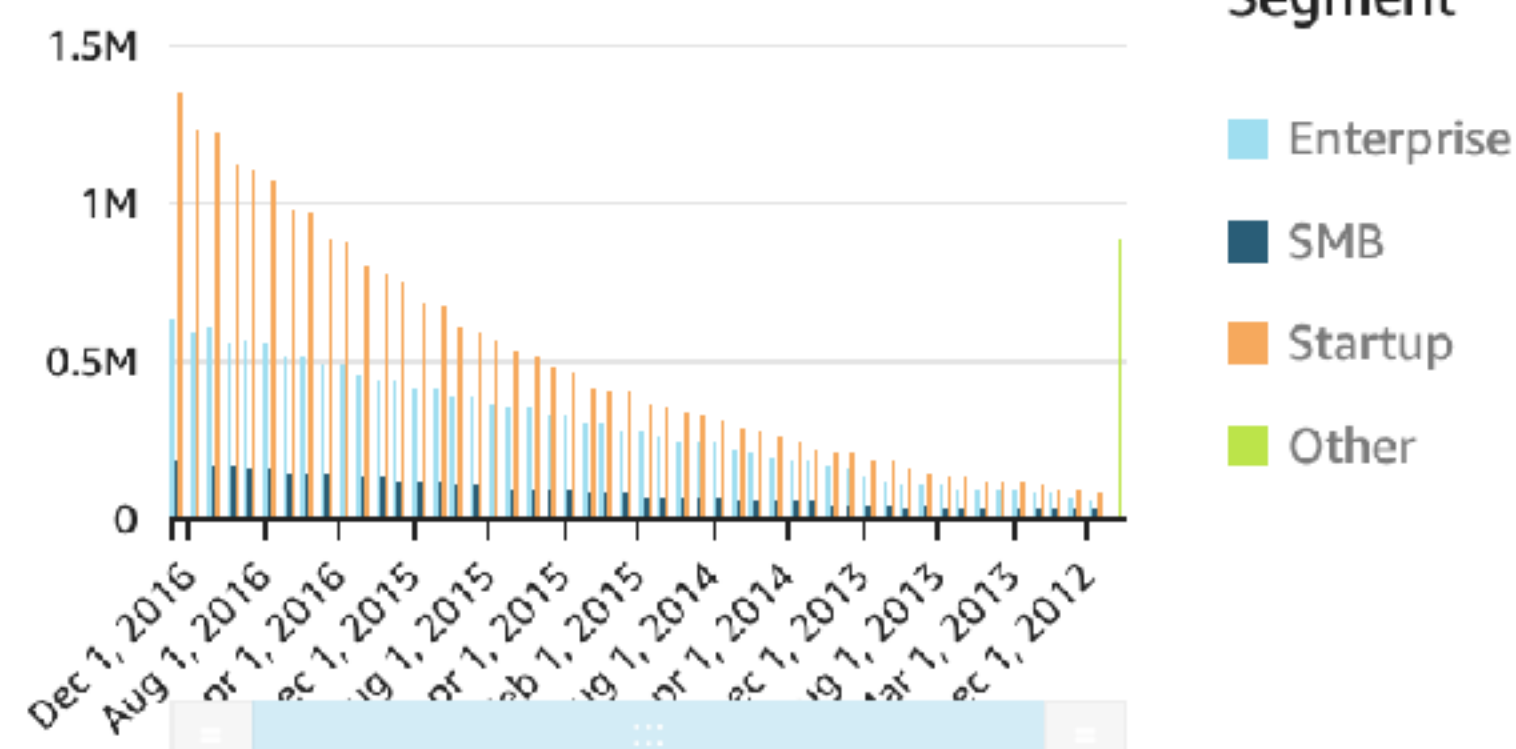


YTD Revenues by Service Line and Segment

Service Line	Billed ...	Customer ID
Billing	17,347,198.35	12,023
HR	17,221,974.97	12,076
Marketing	8,406,810.21	8,950

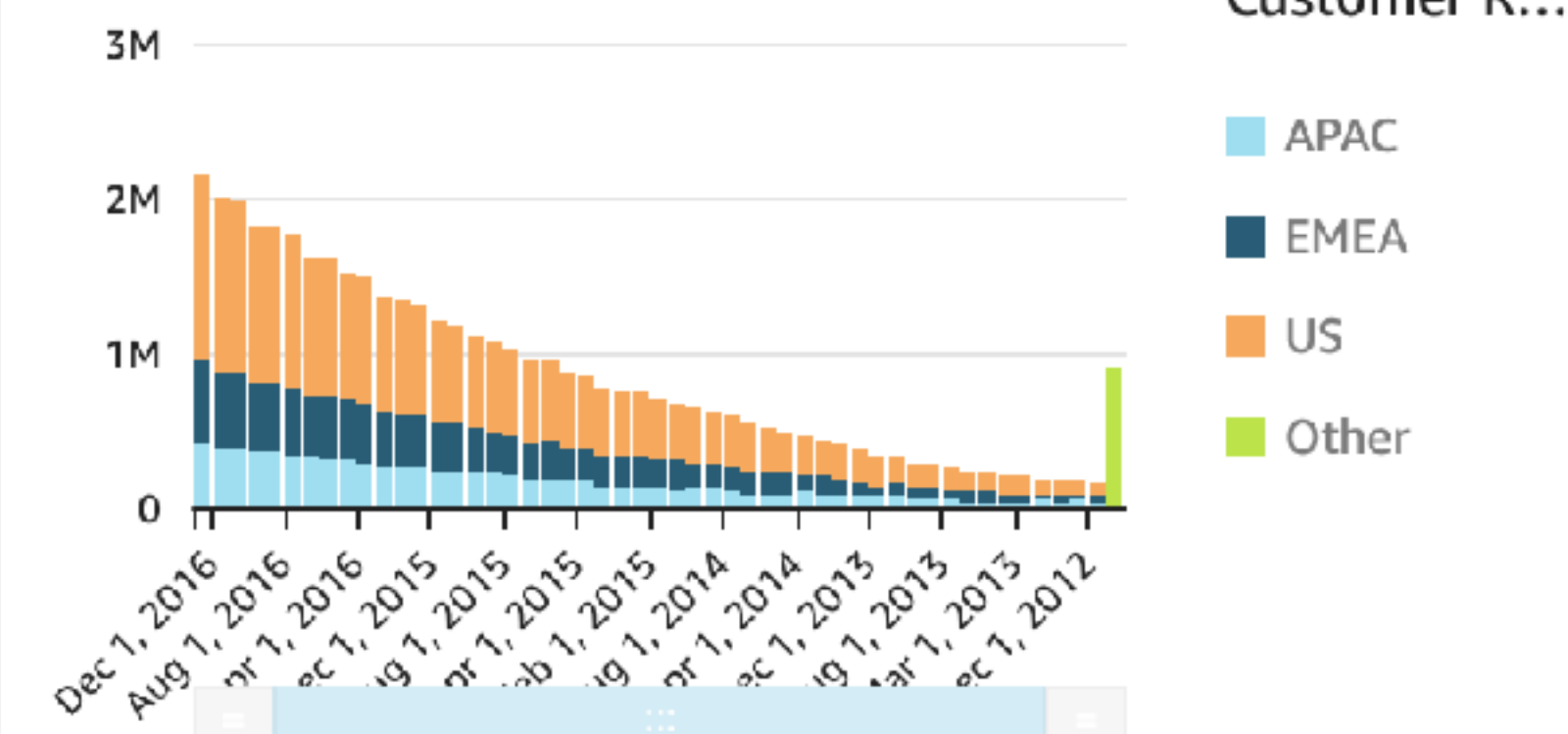
Revenues by Customer Segment

SHOWING TOP 50 IN DATE AND TOP 4 IN SEGMENT



Revenues by Region

SHOWING TOP 50 IN DATE AND TOP 4 IN CUSTOMER REGION



Channel adoption across Time

Average Revenues vs Costs by Segment

 + Code + Text

Connect  |  Editing 

```
[ ] from google.colab import auth
    auth.authenticate_user()
    print('Authenticated')
```

Authenticated

```
[ ] %load_ext google.colab.data_table
```

How is current inventory distributed over locations?



```
%%bigquery --project datameshexample-fulfillment
SELECT location, sum(available) as available_total
FROM `aggregates.inventory_latest`
group by location
order by location
```

1 to 10 of 11 entries [Filter](#) 

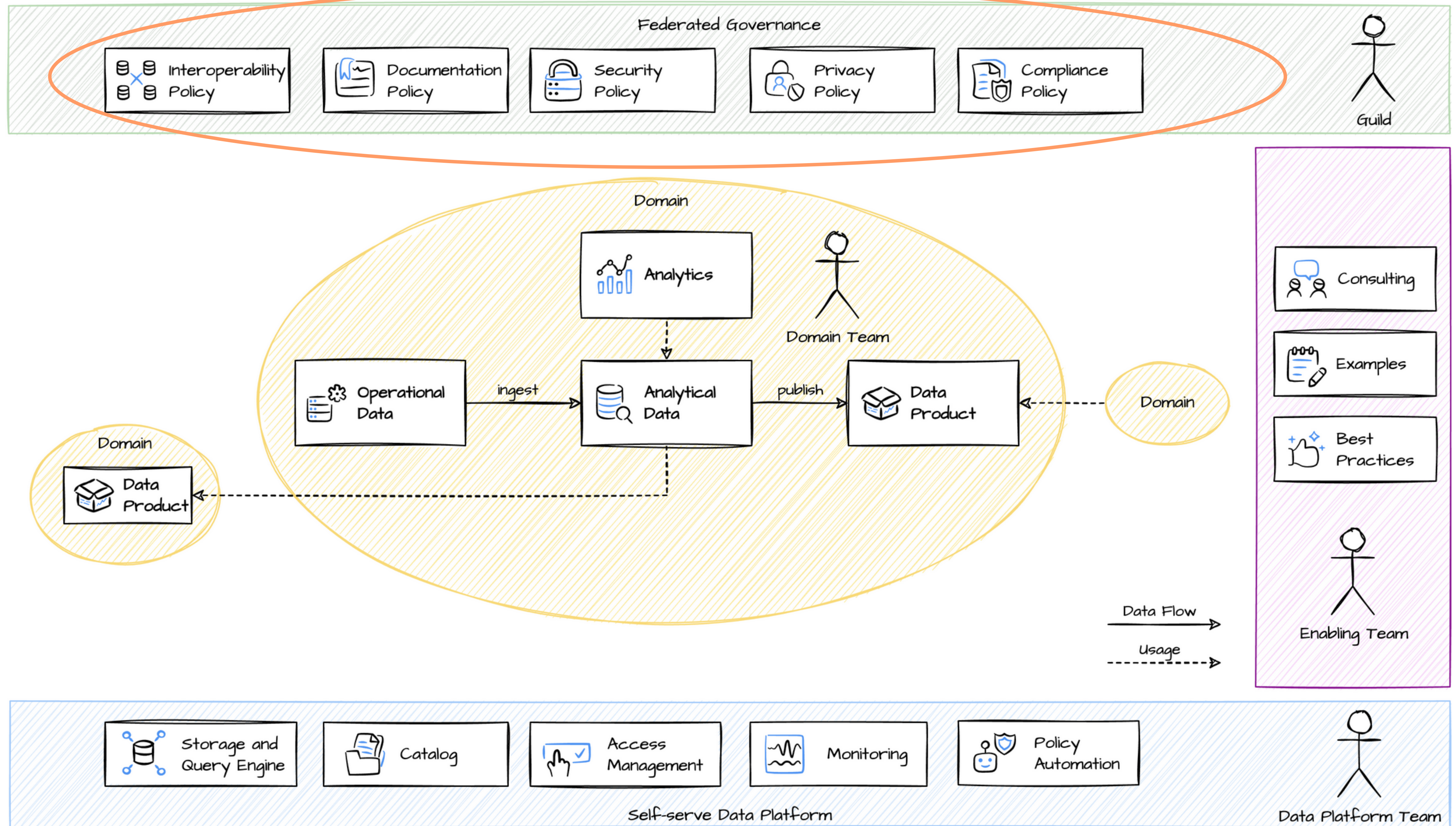
index	location	available_total
0	10	84
1	11	99
2	12	77
3	13	79
4	14	84
5	15	99
6	16	100
7	17	80
8	18	63
9	19	65

Show 10  per page

1

2

Data Mesh Architecture



Data Mesh Governance

Members



Domain Teams
Representatives



Data Platform Team
Representatives



Subject Matter Experts
(legal, privacy, ...) on demand



Global Policies



Context

The issue at hand, goal, situation, former decisions. Why are we talking?



Decision

The agreed decision.



Consequences

Effects, outcomes, costs, follow-up questions.



Policy Automation



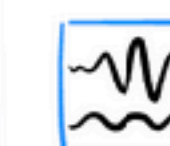
Policy as Code

How to automate a policy
by the data platform?



Automated Tests

Data quality checks
in CI/CD-Pipelines



Monitoring

Compliance and
drift detection

Federated Governance

Global Policy 1: Data Format

CSV-Files

Standard: RFC 4180

Delimiter: COMMA

Header-Record: Required

Global Policy 2: Location

S3

Bucket is managed by domain team

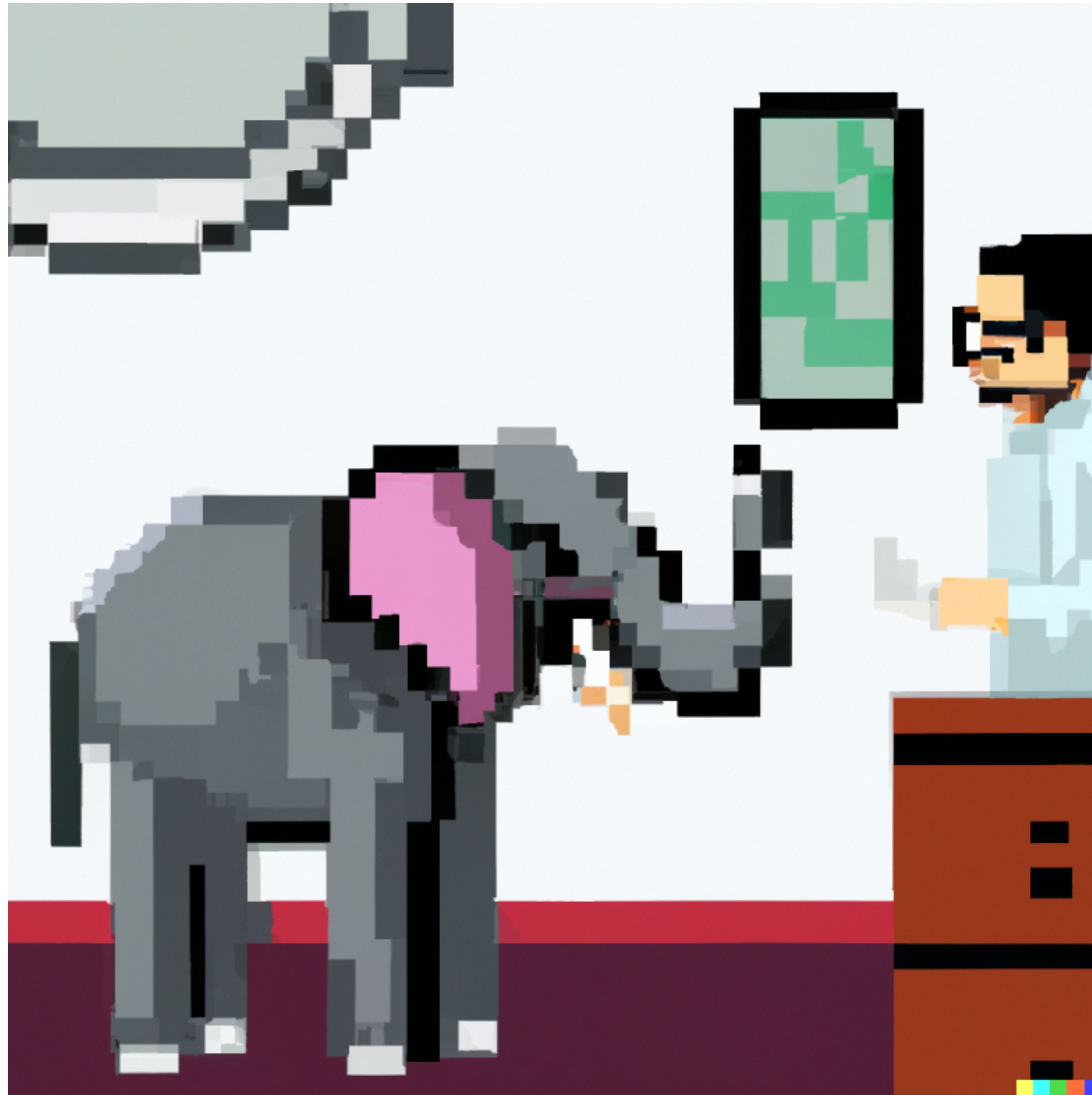
Global Policy 3: Data Discovery

A common table in Wiki that lists:

- Domain Team
- Data Product Name
- Responsible Product Owner
- S3-Location
- Data Freshness / Update frequency
- Tags
- Data Model

Global Policy 4: Access Control

AWS IAM Role based ACL



DALL·E 2: elephant in a room with a software engineer, pixel art

The Elephant in the Room

(The Problem with Data Mesh)

Dev

**Fullstack
Dev**

Fullstack DevOps

Fullstack DevSecOps

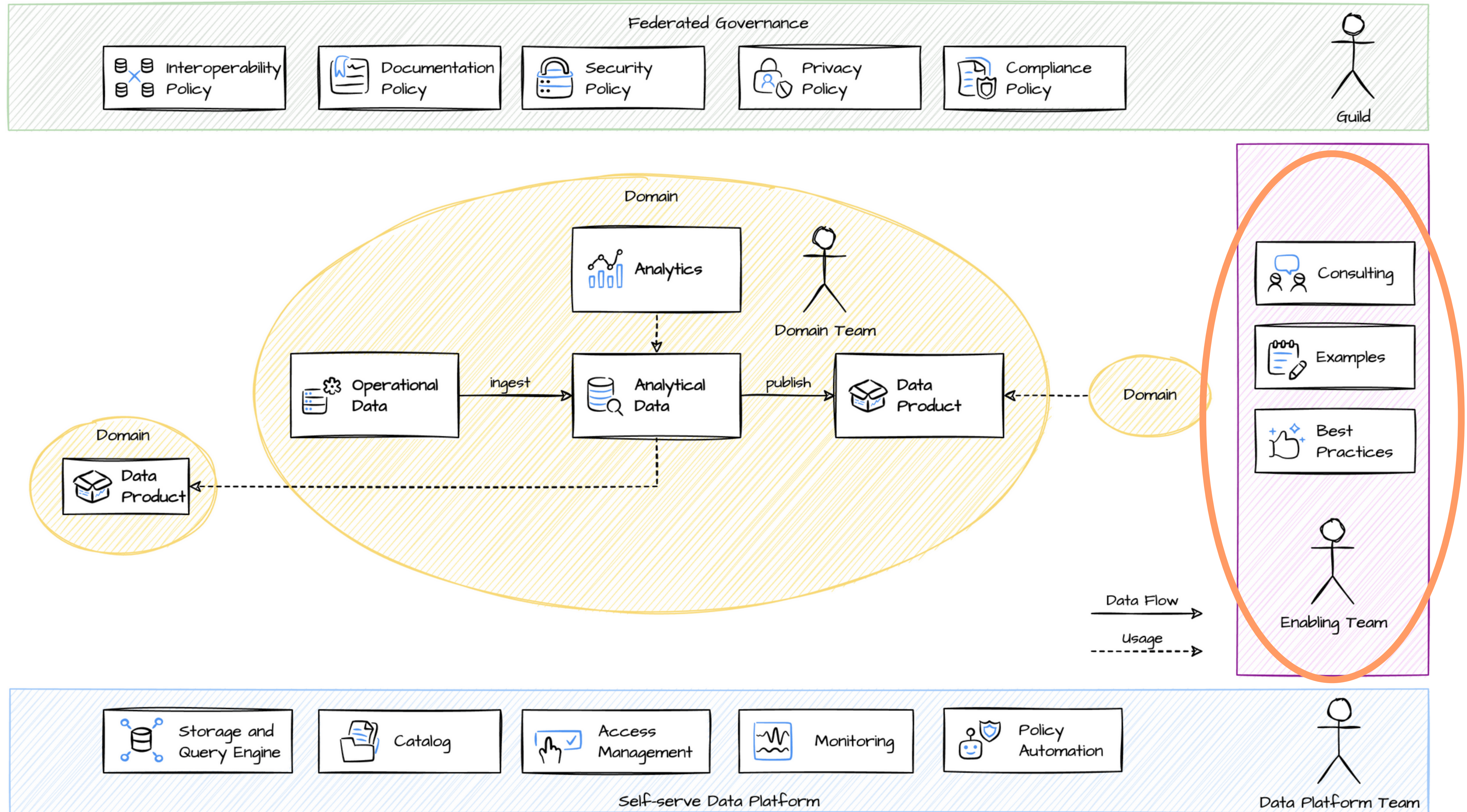
Fullstack
BizDevSecOps

Fullstack
BizDevSecDataOps



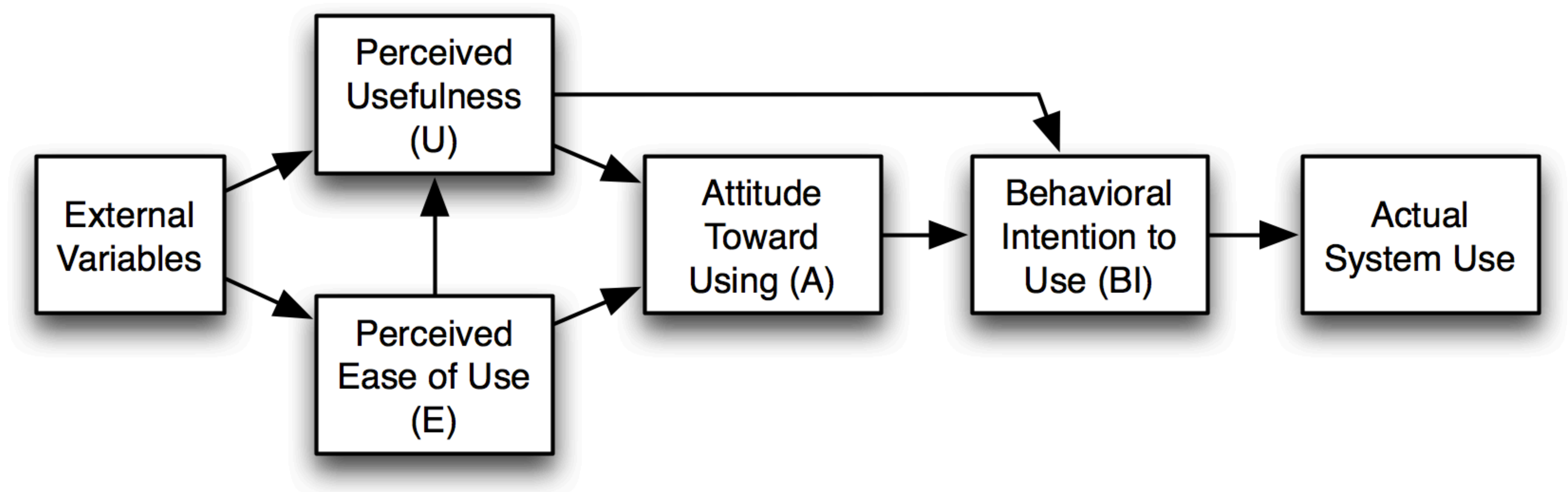
DALL-E 2: A person sits in front of a computer and is exhausted, pixel art

Data Mesh Architecture



Technology Acceptance Model

Enabling Team



Self-Serve
Data Platform

datamesh-architecture.com/fitness-test

www.datamesh-architecture.com/fitness-test

Fitness Test

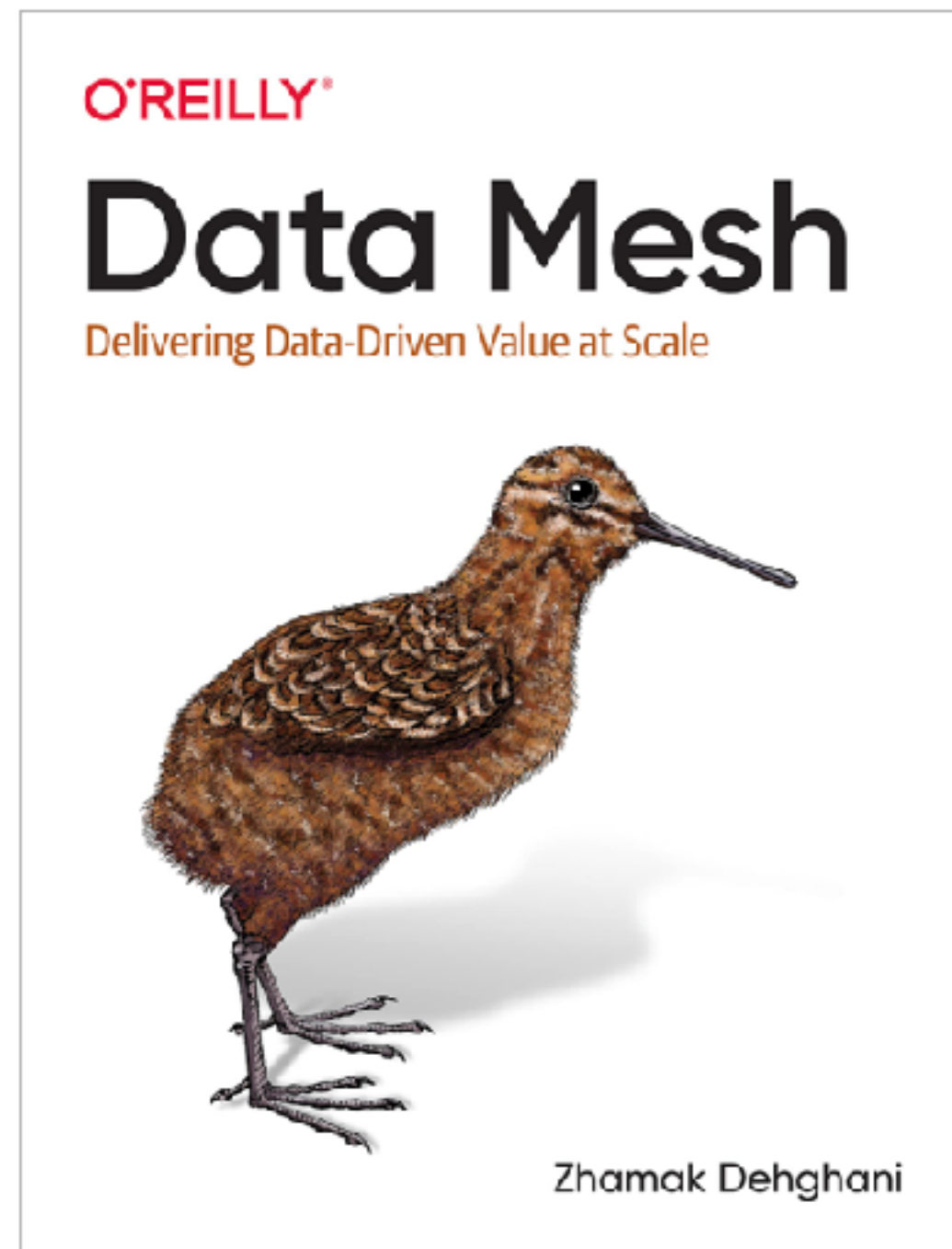
Does Data Mesh fit to me? Am I fit for Data Mesh?

Data Mesh is something new. That's why there are hardly any best practices and experience reports from the industry. Whether data mesh is really a sensible approach for your company can only be determined with a more detailed analysis. But with a few questions you can quickly get an indication in advance. That's why we present here a partial list of heuristics that can help you assess whether Data Mesh makes sense or is at least worth thinking about for your company. We have deliberately avoided a scoring approach, because such a thing always has its weaknesses, and instead worked on an unsorted list of topics with a rough classification of self-assessments into the three response categories "rather not", "worth thinking about" or "useful". This will give you a sense of how ready your company is for Data Mesh.

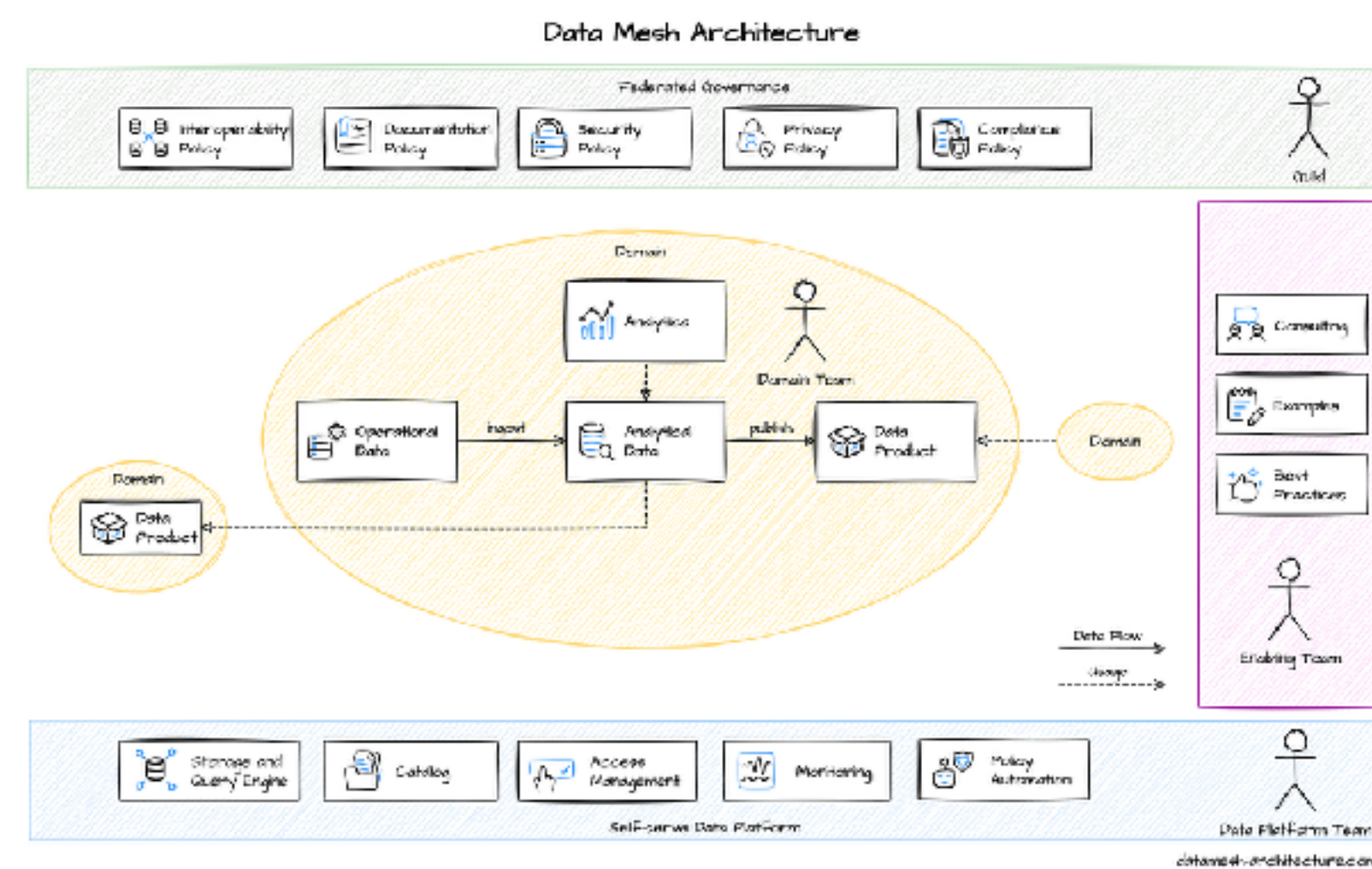
Usage: Go through it row by row. Click on the statement that best represents your company. Your selection is only stored in your local browser.

	<i>Data Mesh might not be the right choice</i>	<i>Data Mesh is worth thinking about</i>	<i>Data Mesh seems useful</i>
Domain-driven Design	Is unknown or does not matter	Partially known and understood	Is understood by all. There are bounded contexts and context maps
Verticalization of the software system based on domains	No. There is a monolith for the entire business	The transformation to microservices has begun	The software system consists of domain-oriented microservices
Number of software development teams	<= 5	> 5	> 10
Make-or-buy strategy for the core software	Standard software (SAP, SaaS, etc.)	Individual software	Individual software
Cloud computing	No issue, many concerns	In the midst of cloud transformation / hybrid cloud	We are cloud native
Team structure	Separate front-end, back-end and operations teams	Mixed teams consisting of specialists	Small cross-functional teams, supported by a platform team

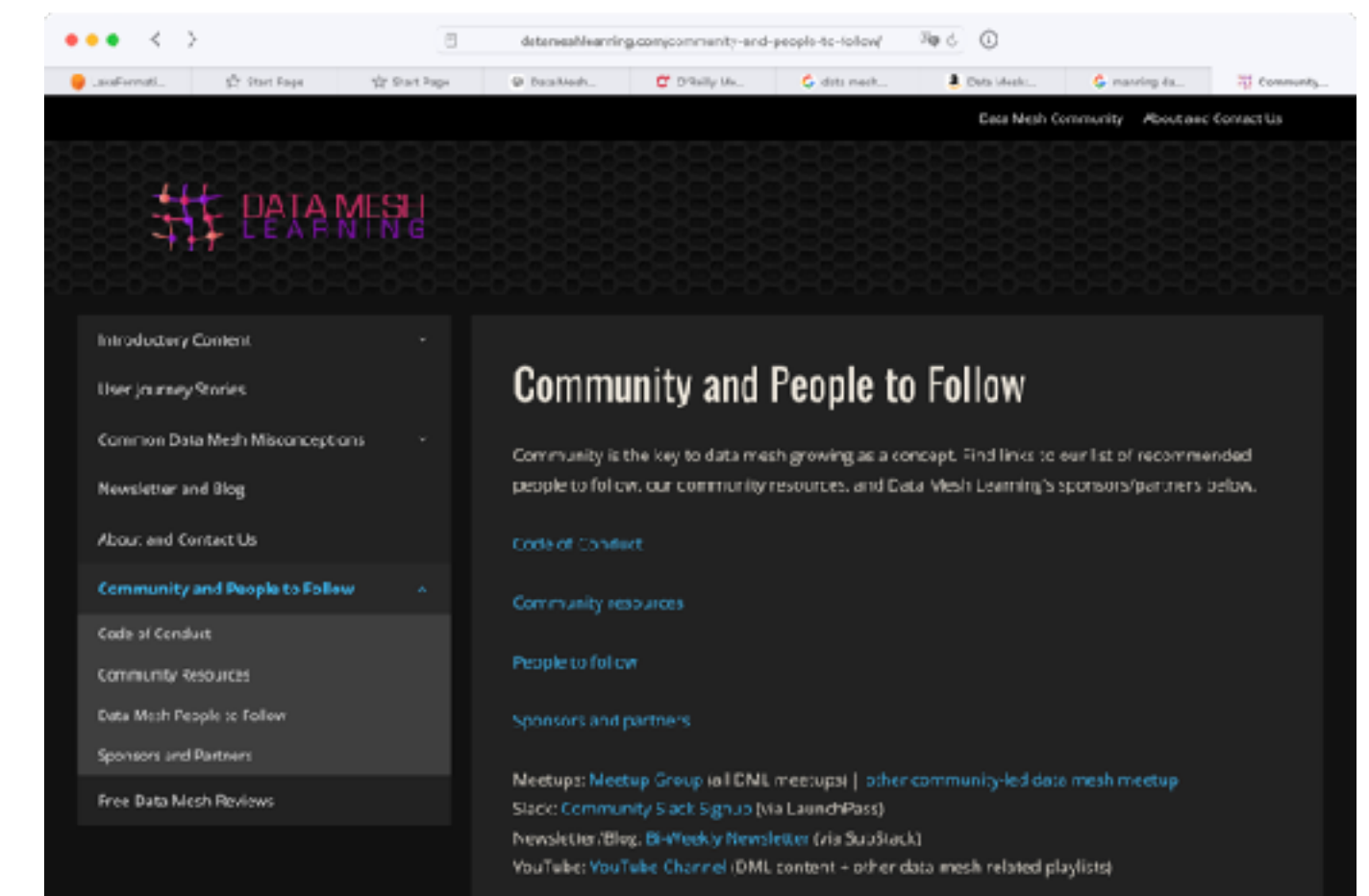
Learn more



<https://www.oreilly.com/library/view/data-mesh/9781492092384/>



datamesh-architecture.com



datameshlearning.com
Podcast, newsletter, slack channel

Data Mesh

Datenanalysen selber machen

INNOQ



JOCHEN CHRIST
@JOCHEN_CHRIST