

Data Contracts

INNOQ



JOCHEN CHRIST
/IN/JOCHENCHRIST

Hi,
I am Jochen

Jochen Christ

Data Mesh Consultant
Product Manager Data Contract Manager

 *Java*

 *Data Mesh*

 *Data-driven Product Development*



datamesh-architecture.com

How To Design a Data Mesh?

A data mesh architecture is a decentralized approach that enables domain teams to perform cross-domain data analysis on their own. At its core is the domain with its responsible team and its operational and analytical data. The domain team ingests operational data and builds analytical data models as data products to perform their own analysis. It may also choose to publish data products with data contracts to serve other domains' data needs.

Data Mesh Architecture

The diagram illustrates the Data Mesh Architecture, showing the flow of data and the roles of different teams and components.

Federated Governance: This layer includes policies such as Interoperability Policy, Documentation Policy, Security Policy, Privacy Policy, and Compliance Policy, managed by a **Governance Group**.

Domain: A central domain team ingests **Operational Data** and publishes **Data Products**. These products are used for **Analytics** and are published as **Data Contracts**. The domain team also uses data products from other domains.

Self-serve Data Platform: This layer includes components like Storage and Query Engine, Data Product Catalog, Data Contract Management, Monitoring, and Policy Automation, managed by the **Data Platform Team**.

Enabling Team: This team provides support through Consulting, Examples, and Best Practices.

Data Flow: Solid arrows represent data flow, and dashed arrows represent usage.

datamesh-architecture.com

O'REILLY®

Deutsche
Ausgabe

Data Mesh

Eine dezentrale Datenarchitektur entwerfen

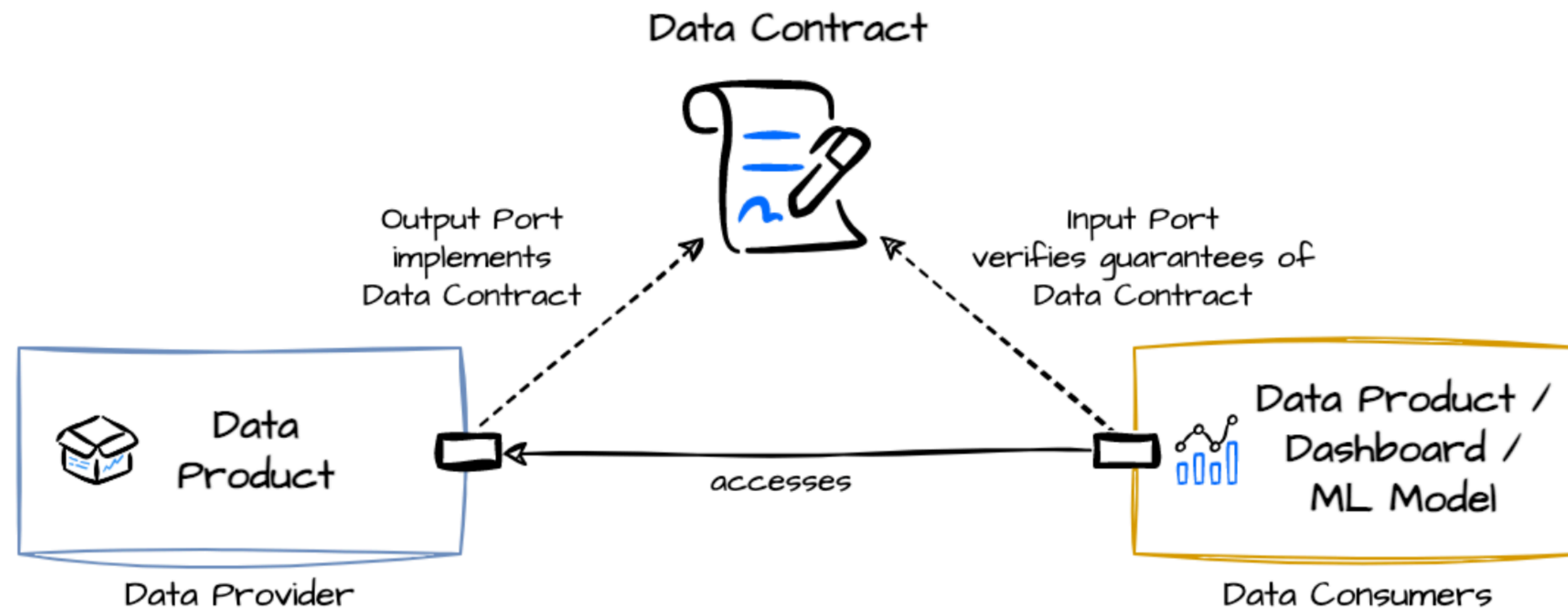


Zhamak Dehghani

Vorwort von Martin Fowler

Übersetzung von Jochen Christ und Simon Harrer

Data Contract Specification



Data contracts bring data providers and data consumers together.

A *data contract* is a document that defines the structure, format, semantics, quality, and terms of use for exchanging data between a data provider and their consumers. A data contract is implemented by a data product's output port or other data technologies. Data contracts can also be used for the input port to specify the expectations of data dependencies and verify given guarantees.

The *data contract specification* defines a YAML format to describe attributes of provided data sets. It is data platform neutral, yet supports well-known formats to express schemas (e.g., dbt models, JSON Schema, Protobuf, SQL DDL) and quality tests (e.g., SodaCL, SQL queries) to avoid unnecessary abstractions. The data contract specification is an open initiative to define a common data contract format. Think of an [OpenAPI specification](#), but for data sets.

Data Contract CLI | CLI to mar x +

cli.datacontract.com

Data Contract CLI

build passing Stars 323 Slack join chat

The `datacontract` CLI is an open source command-line tool for working with [Data Contracts](#). It uses data contract YAML files to lint the data contract, connect to data sources and execute schema and quality tests, detect breaking changes, and export to different formats. The tool is written in Python. It can be used as a standalone CLI tool, in a CI/CD pipeline, or directly as a Python library.

The diagram illustrates the workflow of the Data Contract CLI. At the center is a box representing a data contract specification file named `datacontract.yaml`. The content of this file is as follows:

```
dataContractSpecification: 0.9.3
id: urn:datacontract:orders-latest
info:
  title: Orders Latest
  version: 1.0.0
models:
  orders:
    type: table
    fields:
      order_id:
        type: text
        format: uuid
```

Arrows indicate the following actions:

- import:** SQL DDL, JSON Schema, Avro, and AWS Data Catalog are imported into the central specification.
- export:** The specification is exported to SQL DDL, HTML, Avro, RDF, dbt, SodaCL, Terraform, and ODCS.
- test:** The specification is tested against various data sources: AWS S3, BigQuery, Azure, Databricks, Snowflake, and Kafka.
- diff:** A circular arrow labeled 'diff' indicates the tool's ability to detect changes in the specification.

github.com/datacontract/cli

github.com/datacontract/datacontract-cli



Home > Data Contracts

Data Contracts

Definitions

Add Data Contract

Owner

Data Product

Sort

<p>Articles Internal</p> <p>Products Passed 1 consumer</p> <p>Current state of all articles</p>	<p>Articles History Internal</p> <p>Products Passed 2 consumers</p> <p>All article snapshots since 2020</p>	<p>Customer Cohorts Restricted</p> <p>Marketing 1 consumer</p> <p>A table with customer cohorts and their properties</p>
<p>Customers History PII Sensitive</p> <p>Checkout</p> <p>All customer states, updated on every modifying event. PII included.</p>	<p>Customers History NPII Restricted</p> <p>Checkout Passed</p> <p>All customer states, updated on every modifying event. PII removed.</p>	<p>Customers Latest PII Sensitive</p> <p>Checkout Passed 1 consumer</p> <p>All customers in their latest state, PII included.</p>
<p>Customers Latest NPII Restricted</p> <p>Checkout Passed</p> <p>All customers in their latest state, PII removed.</p>	<p>Orders PII Sensitive</p> <p>Checkout Passed 1 consumer</p> <p>All order-created events, with PII.</p>	<p>Orders NPII Sensitive</p> <p>Checkout Passed 2 consumers</p> <p>All order-created events, PII removed.</p>
<p>Search Queries All</p> <p>Search Passed 1 consumer</p> <p>All search queries and result sets with PII removed.</p>	<p>Shelf Warmers Internal</p> <p>Fulfillment Passed</p> <p>A list of articles with no sales in last 6 months</p>	<p>Stock Updates Internal</p> <p>Fulfillment Failed 1 consumer</p> <p>All stock updates since 2020</p>

Data Contract

Data Contract

~ API for data

Data Contract

~ OpenAPI for data

dataContractSpecification: 0.9.1

id: web-orders-with-consent-v1

info:

title: Web Orders With Consent V1

version: 1.0.0

description: "All orders made through the web channel.\r\nFiltered for orders where customers have expressed consent for analytical use."

owner: checkout

contact:

url: <https://teams.example.com/datacontracts/web-orders-with-consent-v1>

terms:

usage: "The data can be used for analytical and data science use cases, as the customer has expressed their consent."

limitations: "As the dataset is filtered, these data set cannot be used to aggregate financial KPIs.\r\nNot suited for real-time use cases."

billing: \$1000 per month

noticePeriod: P3M

models:

orders:

type: table

description: A successful sale in the web shop

fields:

order_id:

type: string

description: Primary key of the order

billing_customer_id:

type: string

description: Customer ID of the billing customer

shipment_customer_id:

type: string

description: Customer ID of customer to ship the order to

sold_timestamp:

type: timestamp_tz

description: The timestamp of the final confirmation step in the web form.

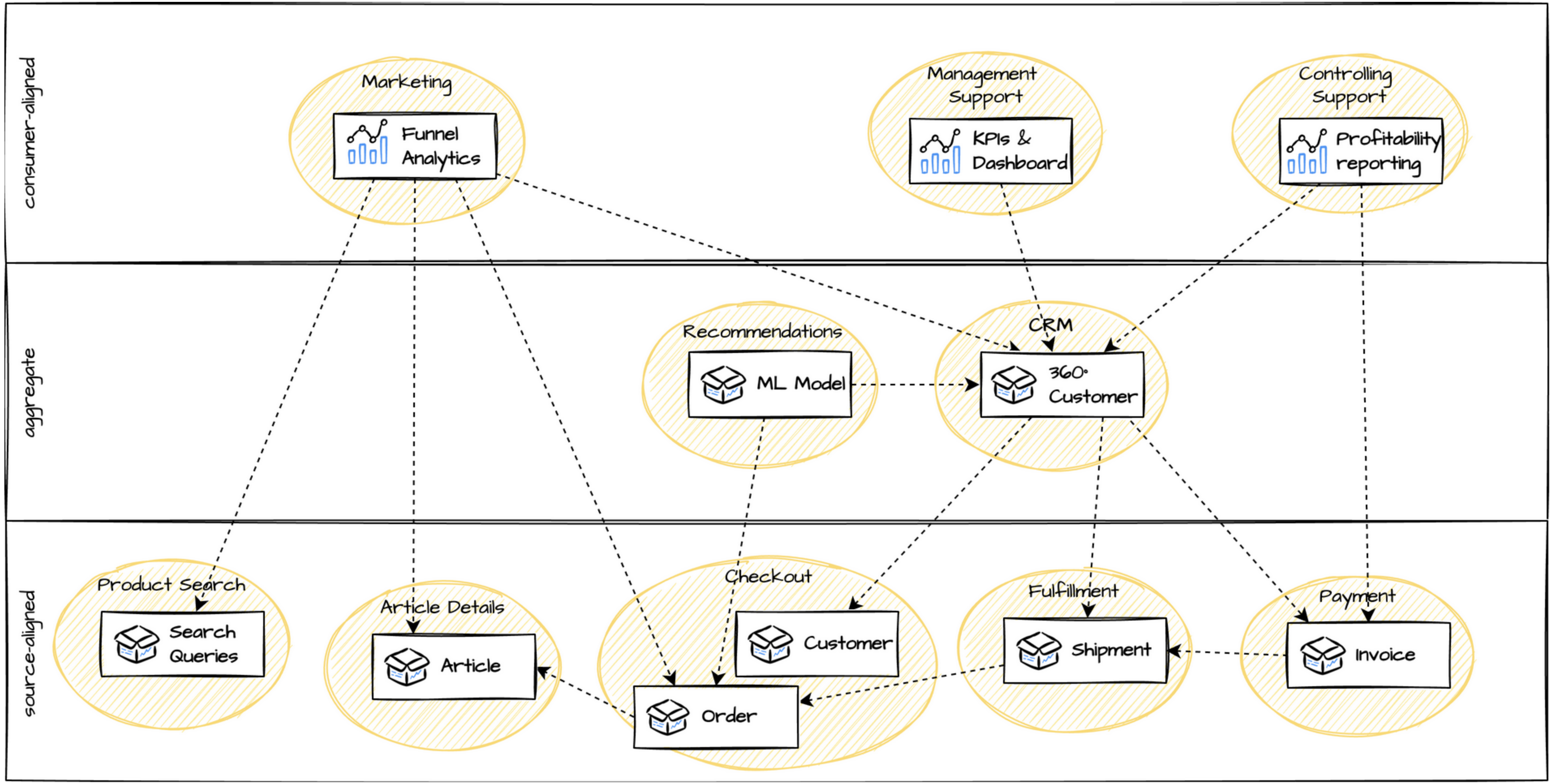
total_amount:

type: bigint

description: The total order amount in the smallest unit of the currency (such as Eurocents)

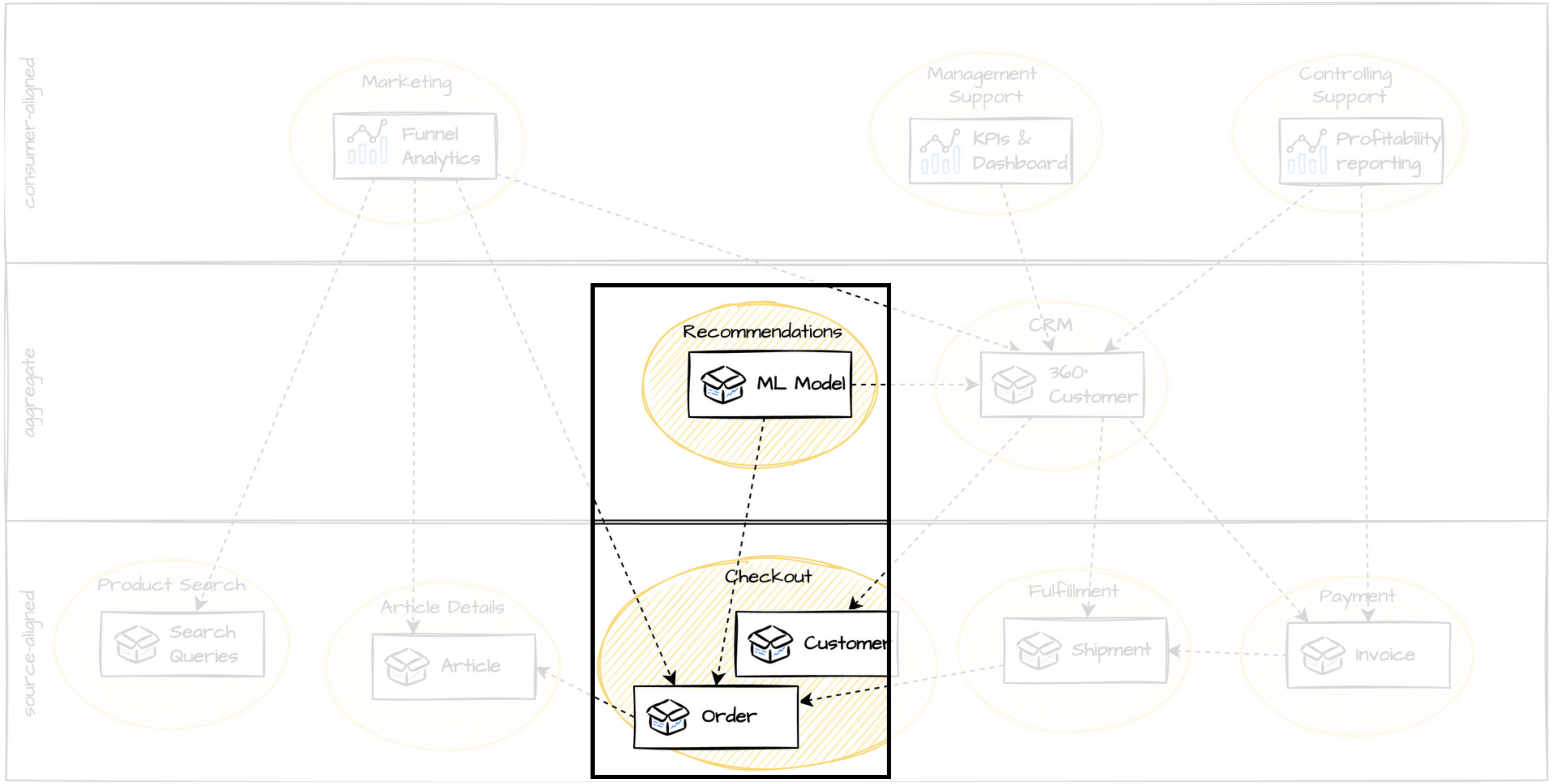
Use cases

- Large datasets ($n > 1$)
- Structured data
 - Tables (BigQuery, Databricks, Iceberg/Delta Files, Postgres, CSV, ...)
 - Objects (JSON/AVRO/Protobuf/XML files in S3 Buckets/SFTP, ...)
 - Events (Kafka, MQ, ...)
 - Usually not for: binary data, reports, ML models, ...
- Read-only
- Shared with other teams
- Usually (but not limited to) for analytical use



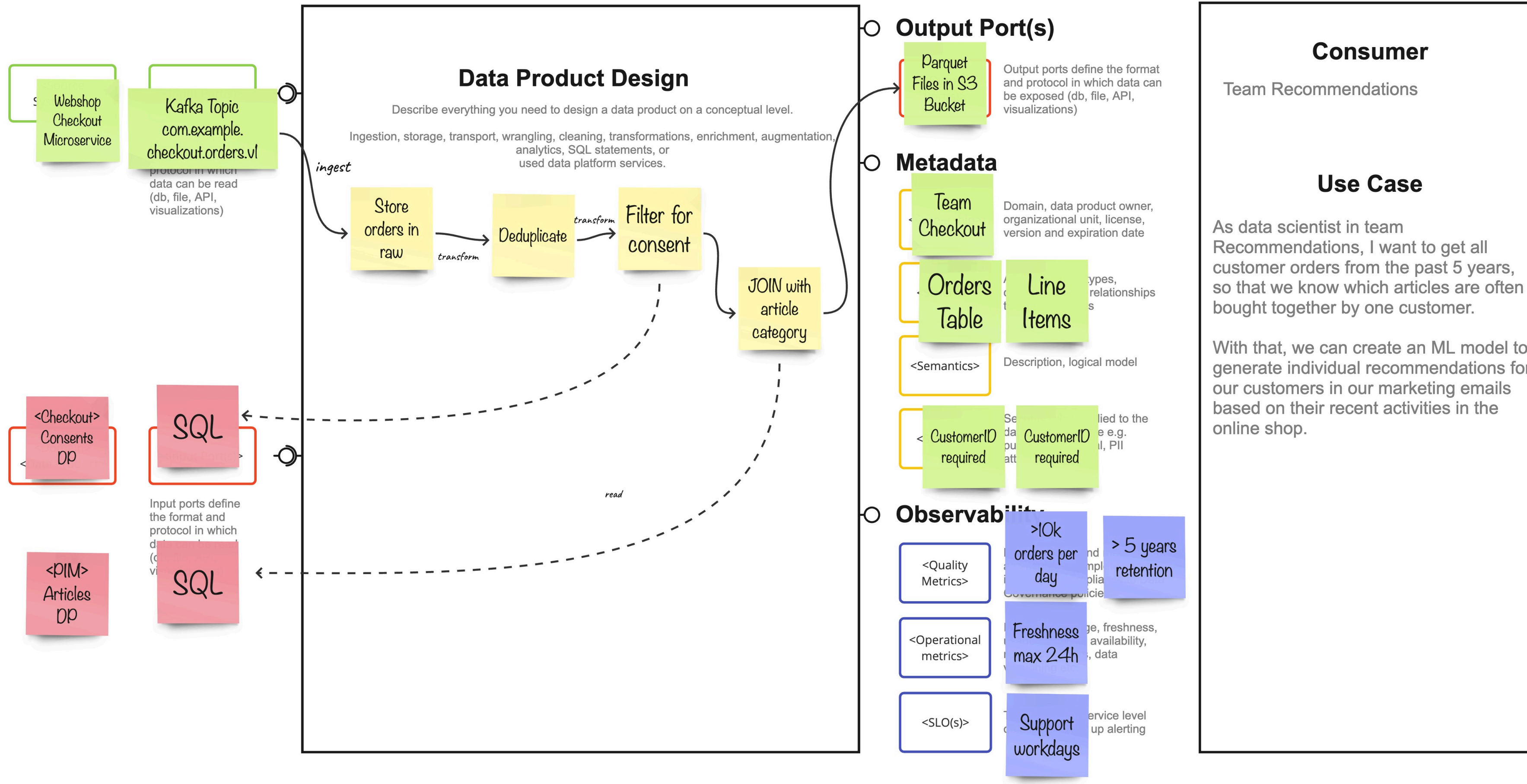
Example

Data Product Orders
for Team Recommendations



Raw Data

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
ws_sold_dat	ws_sold_tim	ws_ship_dat	ws_item_sk	ws_bill_cust	ws_bill_cder	ws_bill_hder	ws_bill_addr	ws_ship_cus	ws_ship_cde	ws_ship_hde	ws_ship_adc	ws_web_pag	ws_web_site	ws_ship_mo	ws_warehou	ws_promo_s	ws_order_nur
2451383	73313	2451482	4591	83074	596485	1096	40907	85919	41329	1140	1351	43	4	4	5	6	1
2451383	73313	2451411	3566	83074	596485	1096	40907	85919	41329	1140	1351	28	7	3	2	271	1
2451383	73313	2451413	7286	83074	596485	1096	40907	85919	41329	1140	1351	58	28	10	5	300	1
2451383	73313	2451393	2755	83074	596485	1096	40907	85919	41329	1140	1351	2	7	6	1	63	1
2451383	73313	2451502	2516	83074	596485	1096	40907	85919	41329	1140	1351	56	16	2	5	18	1
2451383	73313	2451421	16966	83074	596485	1096	40907	85919	41329	1140	1351	50	7	2	2	185	1
2451383	73313	2451457	10402	83074	596485	1096	40907	85919	41329	1140	1351	56	16	16	3	293	1
2451383	73313	2451430	1735	83074	596485	1096	40907	85919	41329	1140	1351	25	19	14	3	202	1
2451383	73313	2451458	15464	83074	596485	1096	40907	85919	41329	1140	1351	13	26	8	5	49	1
2452625	34964	2452675	8925	42296	436090	2684	37278	57428	1447713	3907	30901	12	6	20	2	107	2
2452625	34964	2452716	11041	42296	436090	2684	37278	57428	1447713	3907	30901	55	30	9	1	253	2
2452625	34964	2452715	645	42296	436090	2684	37278	57428	1447713	3907	30901	36	3	6	2	171	2
2452625	34964	2452729	12453	42296	436090	2684	37278	57428	1447713	3907	30901	19	27	18	5	289	2
2452625	34964	2452741	13831	42296	436090	2684	37278	57428	1447713	3907	30901	6	12	19	2	75	2
2452625	34964	2452725	9559	42296	436090	2684	37278	57428	1447713	3907	30901	49	15	11	1	54	2
2452625	34964	2452743	8085	42296	436090	2684	37278	57428	1447713	3907	30901	12	13	5	1	294	2
2452625	34964	2452702	12081	42296	436090	2684	37278	57428	1447713	3907	30901	9	13	16	3	267	2
2452625	34964	2452638	15739	42296	436090	2684	37278	57428	1447713	3907	30901	1	13	16	2	221	2
2451754	1529	2451774	17431	74800	1219525	3450	3375	84180	1487225	5550	41475	14	20	7	5	246	3
2451754	1529	2451819	4694	74800	1219525	3450	3375	84180	1487225	5550	41475	56	11	3	5	237	3
2451754	1529	2451836	2189	74800	1219525	3450	3375	84180	1487225	5550	41475	17	19	1	2	223	3
2451754	1529	2451769	151	74800	1219525	3450	3375	84180	1487225	5550	41475	5	20	6	4	72	3
2451754	1529	2451763	10793	74800	1219525	3450	3375	84180	1487225	5550	41475	44	2	18	2	276	3
2451754	1529	2451863	12782	74800	1219525	3450	3375	84180	1487225	5550	41475	8	26	3	2	94	3
2451754	1529	2451868	15769	74800	1219525	3450	3375	84180	1487225	5550	41475	23	2	9	4	99	3
2451754	1529	2451833	5549	74800	1219525	3450	3375	84180	1487225	5550	41475	5	23	1	5	69	3
2451754	1529	2451858	12475	74800	1219525	3450	3375	84180	1487225	5550	41475	5	23	8	4	23	3
2451754	1529	2451785	5324	74800	1219525	3450	3375	84180	1487225	5550	41475	47	23	20	1	299	3
2451754	1529	2451839	15686	74800	1219525	3450	3375	84180	1487225	5550	41475	47	26	17	2	7	3



Classification
The nature of the exposed data (source-aligned, aggregate, consumer-aligned)
source-aligned

Ubiquitous Language
Context-specific domain terminology (relevant for Data Product)
Data Products polysemes, which are used to create the current Data Product
Order, Line Items, CustomerID

Drive Data Product Development by Business Case

"As data scientist in team Recommendations, I want to get all webshop orders from the past 5 years, so that we know which articles (categories) are often bought together by one customer.

With that, we can create an ML model to generate individual recommendations for our customers in our marketing emails based on their recent activities in the online shop."



Sophia, Data Scientist

Data Model

```
models:
  orders:
    description: One record per order. Includes cancelled and deleted orders.
    type: table
    fields:
      order_id:
        title: Order ID
        type: text
        format: uuid
        description: An internal ID that identifies an order in the online shop.
        example: 243c25e5-a081-43a9-aeab-6d5d5b6cb5e2
        pii: true
        classification: restricted
        required: true
        unique: true
        primary: true
      order_timestamp:
        description: The business timestamp in UTC when payment was successful.
        type: timestamp
        required: true
        example: "2024-09-09T08:30:00Z"
      order_total:
        description: Total amount the smallest monetary unit (e.g., cents).
        type: long
```


Semantic Definitions

```
definitions:  
  order_id:  
    domain: checkout  
    name: order_id  
    title: Order ID  
    type: string  
    description: An internal ID that identifies an order in the or  
    example: 243c25e5-a081-43a9-aeab-6d5d5b6cb5e2  
    pii: true  
    classification: restricted  
  sku:  
    domain: inventory  
    name: sku  
    title: Stock Keeping Unit  
    type: string  
    example: AC1212ME1  
    description: |  
      A Stock Keeping Unit (SKU) is an internal unique identifier  
      It is typically associated with an article's barcode, such a
```

```
models:  
  orders:  
    type: table  
    description: A successful sale in the web shop  
    fields:  
      order_id:  
        $ref: definitions.yaml#/checkout/order_id  
      billing_customer_id:  
        type: string  
        description: Customer ID of the billing customer  
      sold_timestamp:  
        type: timestamp_tz  
        description: The timestamp of the final confirmation step in the web form  
      total_amount:  
        type: bigint  
        description: The total order amount in the smallest unit of the currency  
      total_currency:  
        type: string  
        description: The ISO 4217 currency code.  
  line_items:  
    type: table  
    description: The articles in the order. One row per article.  
    fields:  
      order_id:  
        $ref: definitions.yaml#/checkout/order_id  
      sku:  
        $ref: definitions.yaml#/fulfillment/sku
```


Quality

order_total:

description: Total amount the smallest monetary unit (e.g., cents).

type: long

required: true

examples:

- 9999

quality:

- type: text

description: 95% of all order total values are expected to be between 10 and 499 EUR.

Quality

order_total:

description: Total amount the smallest monetary unit (e.g., cents).

type: long

required: true

examples:

- 9999

quality:

- type: sql

description: 95% of all order total values are expected to be between 10 and 499 EUR.

query: |

```
SELECT quantile_cont(order_total, 0.95) AS percentile_95
```

```
FROM orders
```

mustBeBetween: [1000, 49900]

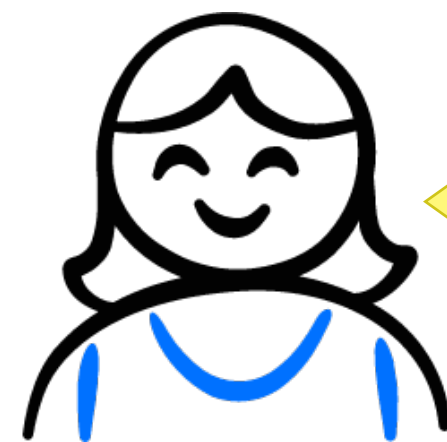
Lineage

```
customer_email_address:  
  description: The email address, as entered by the customer.  
  type: text  
  format: email  
  required: true  
  pii: true  
  classification: sensitive  
  quality:  
    - type: text  
      description: The email address is not verified and may be invalid.  
  lineage:  
    inputFields:  
      - namespace: com.example.service.checkout  
        name: checkout_db.orders  
        field: email_address
```


Drive Data Product Development by Business Case

"As data scientist in team Recommendations, I want to get all webshop orders from the past 5 years, so that we know which articles (categories) are often bought together by one customer.

With that, we can create an ML model to generate individual recommendations for our customers in our marketing emails based on their recent activities in the online shop."



Ah, good to know.
That is OK for us.

Sophia, Data Scientist



"Sorry, we cannot give you **all** orders, but only orders when the customer expressed their consent for analytical use. That are around 80% of all orders.

John, Product Owner Team Checkout

Terms & Conditions

terms:

usage: "The data can be used for analytical and data science use cases, as the customer has expressed their consent."

limitations: |

| As the dataset is filtered, these data set cannot be used to aggregate financial KPIs.

| Not suited for real-time use cases.

billing: \$1000 per month

noticePeriod: P3M

Service Levels

```
servicelevels:  
  availability:  
    description: The server is available during support hours  
    percentage: 99.9%  
  retention:  
    description: Data is retained for one year  
    period: P1Y  
    unlimited: false  
  frequency:  
    description: Data is delivered once a day  
    type: batch  
    cron: 0 0 * * *  
  support:  
    description: The data is available during typical business hours at headquarters  
    time: 9am to 5pm in EST on business days  
    responseTime: 1h  
  backup:  
    description: Data is backed up once a week, every Sunday at 0:00 UTC.  
    interval: weekly  
    cron: 0 0 * * 0  
    recoveryTime: 24 hours  
    recoveryPoint: 1 week
```


Servers (Physical Endpoints)

```
servers:  
  production:  
    type: BigQuery  
    project: acme_sales_prod  
    dataset: orders_latest_pii_v1
```

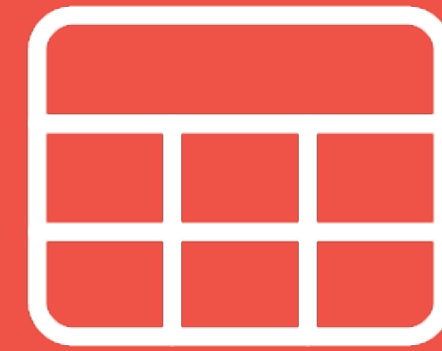
Data Contract



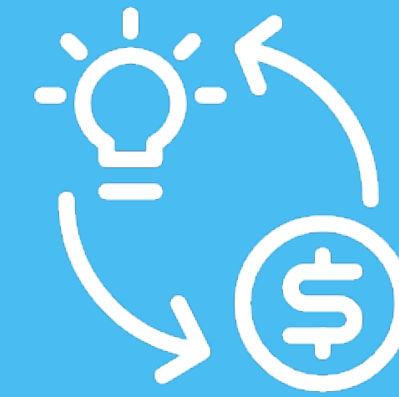
Ownership



Terms



Model



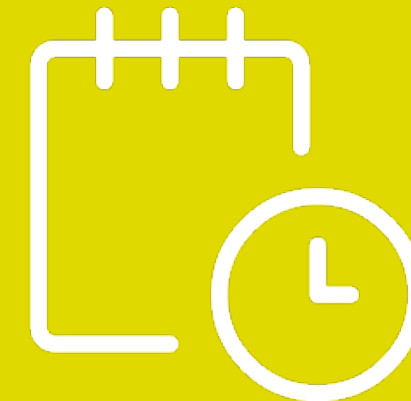
Definitions



Examples



Quality



SLAs



Servers

**A data contract
is not a contract**

A data contract

is not a contract in a legal sense

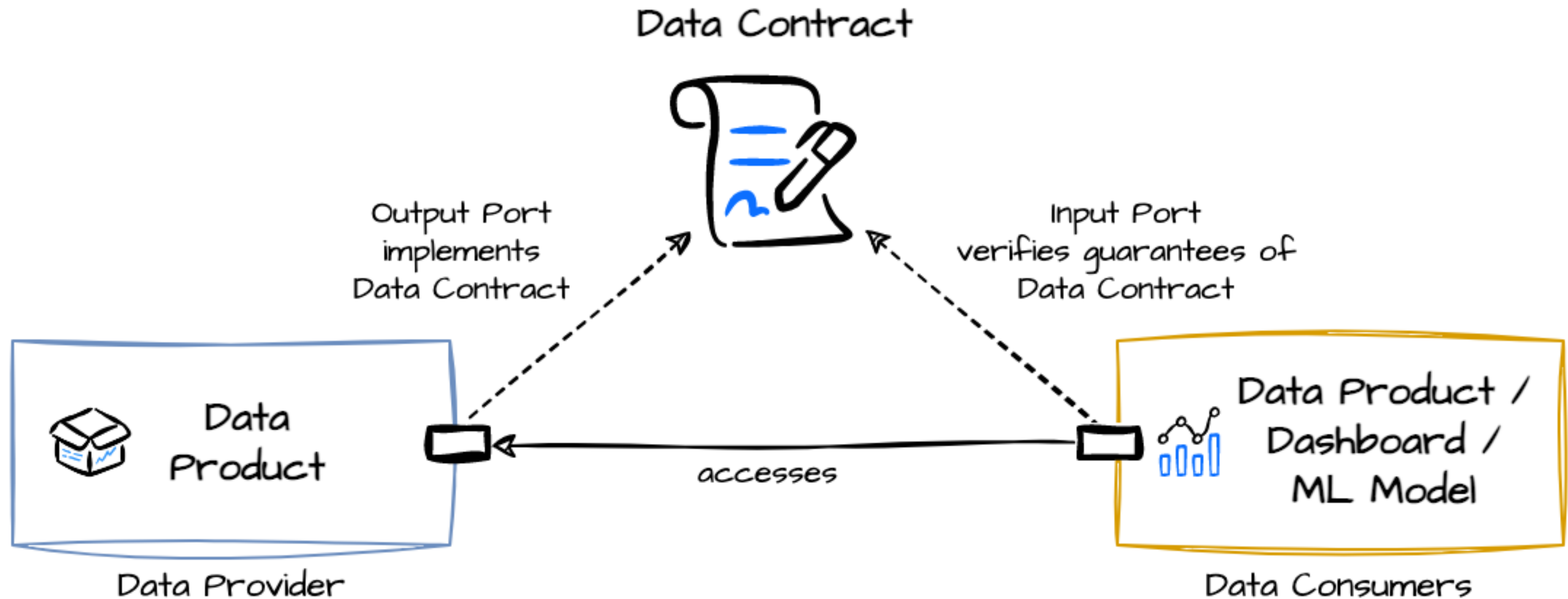
**A data contract
is an offer**

**A data contract
is an offer
for consumers to use data with
guarantees under specified
terms and conditions**

Data Contract

~ API for data

A data contract is implemented by a data product



Data Contract as YAML

```
dataContractSpecification: 0.9.1
id: web-orders-with-consent-v1
info:
  title: Web Orders With Consent V1
  version: 1.0.0
  description: "All orders made through the web channel.\r\nFiltered for orders where customers have expressed consent for analytical use."
  owner: checkout
  contact:
    url: https://teams.example.com/datacontracts/web-orders-with-consent-v1
terms:
  usage: "The data can be used for analytical and data science use cases, as the customer has expressed their consent."
  limitations: "As the dataset is filtered, these data set cannot be used to aggregate financial KPIs.\r\nNot suited for real-time use cases."
  billing: $1000 per month
  noticePeriod: P3M
models:
  orders:
    type: table
    description: A successful sale in the web shop
    fields:
      order_id:
        type: string
        description: Primary key of the order
      billing_customer_id:
        type: string
        description: Customer ID of the billing customer
      shipment_customer_id:
        type: string
        description: Customer ID of customer to ship the order to
      sold_timestamp:
        type: timestamp_tz
        description: The timestamp of the final confirmation step in the web form.
      total_amount:
        type: bigint
        description: The total order amount in the smallest unit of the currency (such as Eurocents)
```

Data Contract CLI



import

```
dataContractSpecification: 0.9.3
id: urn:datacontract:orders-latest
info:
  title: Orders Latest
  version: 1.0.0
models:
  orders:
    type: table
    fields:
      order_id:
        type: text
        format: uuid
```

datacontract.yaml



export

test



datacontract test

```
$ datacontract test datacontract.yaml
```

order_total:

description: Total amount the smallest monetary unit (e.g., cents)

type: long

required: true

customer_id:

description: Unique identifier for the customer.

type: text

minLength: 10

maxLength: 20

customer_email_address:

description: The email address, as entered by the customer. The email

type: text

format: email

required: true

order_total:

description: Total amount the smallest monetary unit (e.g., cents).

type: long

required: true

examples:

- 9999

quality:

- type: sql

description: 95% of all order total values are expected to be between 10 and 499 EUR.

query: |

SELECT quantile_cont(order_total, 0.95) AS percentile_95

FROM orders

mustBeBetween: [1000, 49900]

Data Contract Testing

```
jochen — -zsh
[jochen@Jochens-MacBook-Pro-2 ~ % datacontract test https://datacontract.com/examples/orders-latest/datacontract.yaml
Testing https://datacontract.com/examples/orders-latest/datacontract.yaml
```

Result	Check	Field	Details
passed	Check that JSON has valid schema	orders	All JSON entries are valid. All JSON entries are valid.
passed	Check that JSON has valid schema	line_items	
passed	Check that field order_id is present	orders	
passed	Check that field order_timestamp is present	orders	
passed	Check that field order_total is present	orders	
passed	Check that field customer_id is present	orders	
passed	Check that field customer_email_address is present	orders	
passed	Check that field processed_timestamp is present	orders	
passed	row_count >= 5	orders	
passed	Check that required field order_id has no null values	orders.order_id	
passed	Check that unique field order_id has no duplicate values	orders.order_id	
passed	duplicate_count(order_id) = 0	orders.order_id	
passed	Check that required field order_timestamp has no null values	orders.order_timestamp	
passed	Check that required field order_total has no null values	orders.order_total	
passed	Check that required field customer_email_address has no null values	orders.customer_email_address	
passed	Check that required field processed_timestamp has no null values	orders.processed_timestamp	
passed	Check that field lines_item_id is present	line_items	
passed	Check that field order_id is present	line_items	
passed	Check that field sku is present	line_items	
passed	values in (order_id) must exist in orders (order_id)	line_items.order_id	
passed	row_count >= 5	line_items	
passed	Check that required field lines_item_id has no null values	line_items.lines_item_id	
passed	Check that unique field lines_item_id has no duplicate values	line_items.lines_item_id	

```
● data contract is valid. Run 23 checks. Took 6.776398 seconds.
jochen@Jochens-MacBook-Pro-2 ~ %
```


✖ Change column name #11

🔄 Re-run jobs ▾



🏠 Summary

Jobs

✖ checkBreakingChanges

Run details

🕒 Usage

📄 Workflow file

checkBreakingChanges

failed 5 days ago in 7s

🔍 Search logs



- > ✔ Set up job 1s
- > ✔ Run actions/checkout@v4 1s
- > ✔ Get CLI 0s
- ▼ ✖ Check backwards compatibility 0s

```
1 ▶ Run ./datacontract breaking --with https://raw.githubusercontent.com/datacontract/cli-examples/main/datacontract.yaml
```

```
4 Found 1 differences between the data contracts!
```

```
5
```

```
6 ● Difference 1:
```

```
7 Description: field 'my_table.my_column' was removed
```

```
8 Type: field-removed
```

```
9 Severity: breaking
```

```
10 Level: field
```

```
11 Model: my_table
```

```
12 Field: my_column
```

```
13 Exiting application with error: found breaking differences between the data contracts
```

```
14 Error: Process completed with exit code 1.
```

datacontract export

```
cli — -zsh
(venv) jochen@Jochens-MacBook-Pro-2 cli % datacontract export --format sql
https://datacontract.com/examples/orders-latest/datacontract.yaml

-- Data Contract: urn:datacontract:checkout:orders-latest
-- SQL Dialect: snowflake
CREATE TABLE orders (
  order_id TEXT not null primary key,
  order_timestamp TIMESTAMP_TZ not null,
  order_total NUMBER not null,
  customer_id TEXT,
  customer_email_address TEXT not null,
  processed_timestamp TIMESTAMP_TZ not null
);
CREATE TABLE line_items (
  lines_item_id TEXT not null primary key,
  order_id TEXT,
  sku TEXT
);
(venv) jochen@Jochens-MacBook-Pro-2 cli %
```

Type
html
jsonschema
odcs
sodacl
dbt
dbt-sources
dbt-staging-sql
rdf
avro
protobuf
terraform
sql
sql-query
great-expectations
bigquery
go
pydantic-model
DBML

datacontract export --format html

Data Model
The logical data model

orders table
One record per order. Includes cancelled and deleted orders.

Order ID <code>order_id</code>	text	An internal ID that identifies an order in the online shop. <small>primary required unique format:uuid restricted PII</small>
<code>order_timestamp</code>	timestamp	The business timestamp in UTC when the order was successfully registered in the source system and the payment was successful. <small>required</small>
<code>order_total</code>	long	Total amount the smallest monetary unit (e.g., cents). <small>required</small>
<code>customer_id</code>	text	Unique identifier for the customer. <small>minLength:10 maxLength:20</small>
<code>customer_email_address</code>	text	The email address, as entered by the customer. The email address was not verified. <small>required format:email sensitive PII</small>
<code>processed_timestamp</code>	timestamp	The timestamp when the record was processed by the data platform. <small>required</small>

line_items table
A single article that is part of an order.

datacontract catalog

The screenshot shows a web browser window with the following elements:

- Browser Tab:** "Data Contract" with a close button (x) and a plus sign (+).
- Address Bar:** "datacontract.com/examples/index.html" with navigation icons (back, forward, refresh, home) and utility icons (star, lock, clipboard, profile, menu).
- Page Header:** "Data Contracts" on the left and "datacontract.com" on the right.
- Main Section:** "Data Contract Catalog" with a search bar containing the text "Search".
- Grid of Data Contracts:**
 - Orders Latest (Nested):** Checkout Team. Description: "Successful customer orders in the webshop. All orders since 2020-01-01. Orders with their line items are in their current state (no history included)." Includes a document icon.
 - Orders Latest:** Checkout Team. Description: "Successful customer orders in the webshop. All orders since 2020-01-01. Orders with their line items are in their current state (no history included)." Includes a document icon.
 - COVID-19 cases:** Johns Hopkins University Consolidated data on COVID-19 cases, sourced from Enigma. Includes a document icon.
 - Muellimperium Exchange Format:** Emperor of the Muellimperium. Description: "The Muellimperium Exchange Format is a data contract for exchanging data between the Muellimperium and its partners." Includes a document icon.
- Footer:** "Created at 05 Jun 2024 04:39:48 UTC with Data Contract CLI v0.10.7" and "Supported with ❤️ by Data Mesh Manager".
- Page Navigation:** "Data Contract CLI" and "Data Contract Specification" links.

Data Contract CLI



import

```
dataContractSpecification: 0.9.3
id: urn:datacontract:orders-latest
info:
  title: Orders Latest
  version: 1.0.0
models:
  orders:
    type: table
    fields:
      order_id:
        type: text
        format: uuid
```

datacontract.yaml



export

test



Enterprise Data Marketplace **(Commercial Tooling)**

Browser: Data Contract Manager | localhost:8080/demo129500424135/datacontracts

Navigation: Search | **Data Contracts** | Data Products | Data Governance AI | More

ACME | User Profile

Data Contracts

Search [] | Owner [] | Data Product [] | Sort []

Add Data Contract

Contract Name	Owner	Data Product	Access	Status	Consumers	Description
Articles	Products	Internal	Internal	Passed	1 consumer	Current state of all articles
Articles History	Products	Internal	Internal	Passed	2 consumers	All article snapshots since 2020
Customer Cohorts	Marketing	Restricted	Restricted		1 consumer	A table with customer cohorts and their properties
Customers History	Checkout	PII	Sensitive			All customer states, updated on every modifying event. PII included.
Customers History NPII	Checkout	Restricted	Restricted	Passed		All customer states, updated on every modifying event. PII removed.
Customers Latest	Checkout	PII	Sensitive	Passed	1 consumer	All customers in their latest state, PII included.
Customers Latest NPII	Checkout	Restricted	Restricted	Passed		All customers in their latest state, PII removed.
Orders	Checkout	PII	Internal	Passed	1 consumer	All order-created events, with PII.
Orders NPII	Checkout	Sensitive	Sensitive	Passed	2 consumers	All order-created events, PII removed.
Search Queries All	Search	Passed		Passed	1 consumer	All search queries and result sets with PII removed.
Shelf Warmers	Fulfillment	Internal	Internal	Passed		A list of articles with no sales in last 6 months
Shipments	Checkout	restricted	restricted			This data can be used for analytical purposes

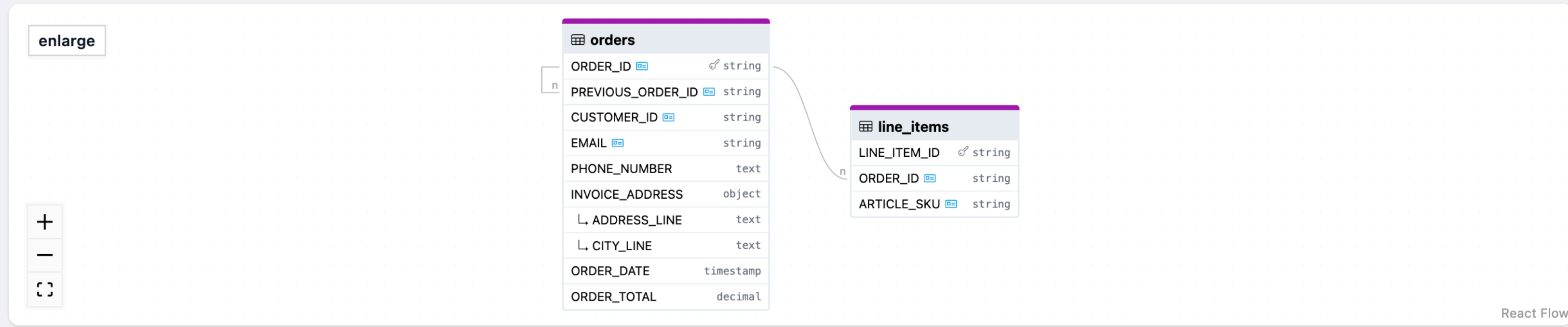
Home > Data Contracts > Orders

Orders

snowflake_orders_pii_v2 1.0.0

Edit YAML Edit Request Access

Checkout active PII Sensitive Data Contract Specification 0.9.3



Info


Information about the data contract

Title	Orders	Version	1.0.0
Description	All order-created events, with PII.		
Owner	Checkout	Contact	Osbert Mackenzie ✉ osbert.mackenzie@demo1ICjy6gRN5HYRuhLmASS...

Terms Data Contract
Terms and conditions of the data contract

Implementation

Data product output port implementing this contract

 **Orders**
snowflake_orders_pii_v2

Passed Passed 5 checks
2 minutes ago

Data Governance AI


Automated policy checks

Ownership ✓
The data contract conforms to the policy

Request Access


You are requesting access to the data product **Orders** on output port **snowflake_orders_npii_v2**.
The system will create a data usage agreement for the team **Checkout** to approve.

Consumer Required




Data Product

Request access for one of your data products.



Team

Request access for everybody in your domain team.



User

Request access for yourself.

Consumer Data Product Required

Recommendations ML Model (Team Recommendations) ▼

Select your data product that needs to access and use the provided data.

Purpose Required

We need to get all webshop orders from the past 5 years so that we know which articles (categories) are often bought together by one customer. With that, we can create an ML model to generate individual recommendations for our customers in our marketing emails based on their recent activities in the online shop. ✓

Why do you want access and what do you want to do with the data?

Approve Access Request

Team **Recommendations** requests access to data product **Orders**.

As data product owner, you can approve or reject this request to grant access to your data product.

Approve

Reject

Data Platform

Status of the data platform integration

Status

 Permissions granted

Updated

1 minutes ago

Role

agreement_55gCnBIHYdUWnUJJKYbYIs_role

Agent

Data Mesh Manager Platform Agent v0.1

Details

```
CREATE ROLE agreement_55gCnBIHYdUWnUJJKYbYIs_role;  
GRANT ROLE op_orders_snowflake_orders_npii_v2_role TO ROLE  
agreement_55gCnBIHYdUWnUJJKYbYIs_role;  
GRANT ROLE agreement_55gCnBIHYdUWnUJJKYbYIs_role TO ROLE ;
```

Know Your Consumers (KYC)

localhost:8080/demo129500424135/dataproducts/orders

Search Data Contracts Data Products Data Governance AI More

ACME

Data Products > Orders

Orders

orders

Checkout source-aligned active managed demo

Edit YAML Edit Request Access

React Flow

Info

Information about the data product

Name	ID
Orders	orders
Description	Successful customer orders in the webshop. All orders since 2020-01-01.
platformRole	

Data Product Controlling

Monitor business value, costs, and compliance

Consumers	3
Costs	\$6,700.00

- Data product owners **know their consumers**
- **Access** is fully automated
- **Life-Cycle Management:** Access can be cancelled by consumer or providers
- Important for data product **evolution** (e.g. breaking changes)

Data Governance ✨
(Commercial Tooling)

Personal Identifiable Information (PII)

Edit

GLOBAL-5 Accepted

In a data contract, data model fields that are potential PII candidates must have the pii flag set to true or false based on the domain context.

Typical PII candidates are (but not limited to these):

- first name
- last name
- home address
- email address
- telephone number
- passport number
- social security number
- photo of face
- credit card number
- account user name
- financial records
- medical records

Data Governance AI

Preview

Data Governance AI can make mistakes. Check important results. Data is not used to train models.

Articles



The data contract conforms to the policy

Articles History



Customer Cohorts

1 issues

1. The field 'customer_id' is not flagged with the 'pii' attribute.

Customers History

2 issues

1. The field 'email' is potentially PII but does not have the 'pii' flag set.
2. The field 'customer_id' is potentially PII but does not have the 'pii' flag set.

Customers History NPII



The data contract conforms to the policy

Customers Latest



The data contract conforms to the policy



Data Governance AI

Policy Check



Policy

Personal Identifiable Information (PII)

Data Contract

Articles History

Created At

2024-06-12T03:25:09.078187Z

Checked At

2024-09-16T04:46:46.362832Z (took 5 seconds)

Status

completed

[Rerun](#)

Applicable

true (The policy is applicable to this data contract.)

Adopted

false

Issues

The 'name' field does not have a pii flag set.

Recommendation: Add a pii flag to the 'name' field and set it to true or false based on the domain context.

[Ignore](#)

The 'brand_name' field does not have a pii flag set.

Recommendation: Add a pii flag to the 'brand_name' field and set it to true or false based on the domain context.

[Ignore](#)

Ownership

GLOBAL-1

 Accepted

Data Governance AI

 Checked 12 data contracts



Data Classification

GLOBAL-2

 Accepted

Data Governance AI

 Issues with 4 data contracts



Mandatory fields

GLOBAL-3

 Accepted

Data Governance AI

 Checked 12 data contracts




Snowflake Naming Conventions

GLOBAL-4

 Accepted

Data Governance AI

 Issues with 3 data contracts



Personal Identifiable Informatio...

GLOBAL-5

 Draft

Data Governance AI

 Issues with 4 data contracts



Summary

Data Contract

~ API for data

Data Contract

promotes collaboration

Data Contract

enforces data quality

Data Contract

is not a contract

Links

Open Source

- datacontract.com
- cli.datacontract.com
- editor.datacontract.com

Free

- gpt.datacontract.com

Commercial

- datacontract-manager.com

Data Contracts

INNOQ



JOCHEN CHRIST
@JOCHEN_CHRIST