Data Mesh & Data Vault @ Deichmann

27.11.2025

Das sind wir!

Anh Le
Data Engineer
IT Data & Analytics

- DataOps Advocate
- data4good Volunteer
- Hobby: Musicals & Theater

Sabine Siebenmorgen
Data Engineer
IT Data & Analytics

- Agile Mindset Driver
- Databricks Crafter
- DV2 Modelling Enthusiast
- Hobbies: photography, board games

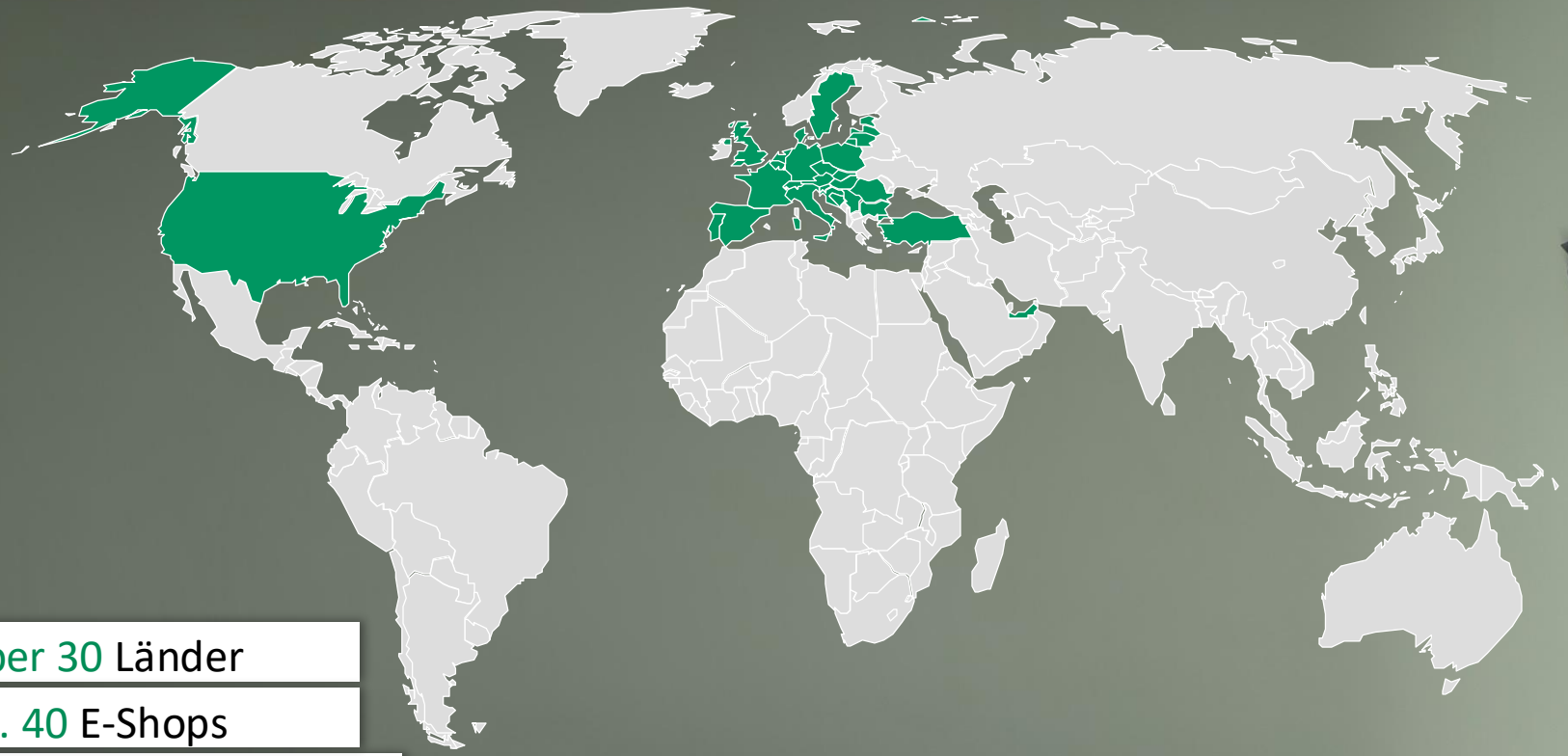
Sebastian Ennigkeit
Developer
IT Data & Analytics

- DV2 Practitioner
- Agile Enthusiast
- Hobbies: Music, Travel, Basketball

Kevin Haferkamp
Head of Data Platform
& Engineering CoE
IT Data & Analytics

- DV2 Practitioner
- Data Modelling Afficionado
- Functional Programming
- Hobbies: JPKR

Nr. 1 in Europa



- über 30 Länder
- ca. 40 E-Shops
- ca. 4.700 Filialen
- ca. 49.900 Mitarbeitende
- über 180.000.000 Paar Schuhe
- ca. 8.700.000.000 Euro Umsatz

DEICHMANN GROUP

DEICHMANN · DOSENBACH · vanHaren · OCHSNER SHOES · OCHSNER SPORT · BACK TO BLACK SHOES · Snipes · solebox. · supremo · urbanstyles.

DEICHMANN: Ein Familienunternehmen

Heinrich Deichmann († 1940)

- Gründete das Unternehmen 1913 im Ruhrgebiet
- Startete mit einem Schuhmacherladen im Arbeiterbezirk Essen-Borbeck

Dr. Heinz-Horst Deichmann († 2014)

- 1956 übernahm der Sohn des Gründers die Leitung der Firma mit Filialen in Düsseldorf und Oberhausen
- Baute das Unternehmen zu Europas größtem Schuheinzelhändler auf

Heinrich Deichmann

- Der Enkel des Gründers leitet das Unternehmen seit 1999
- Unter seiner Leitung werden die internationale Expansion, die Digitalisierung und der Omnichannel-Commerce stetig voran getrieben
- Darüber hinaus setzte er im Einkauf konsequent die Vertikalisierung der Beschaffung um

Samuel Deichmann

- Seit 2020 ist auch der Sohn von Heinrich Deichmann und Urenkel des Gründers Mitglied der Geschäftsleitung

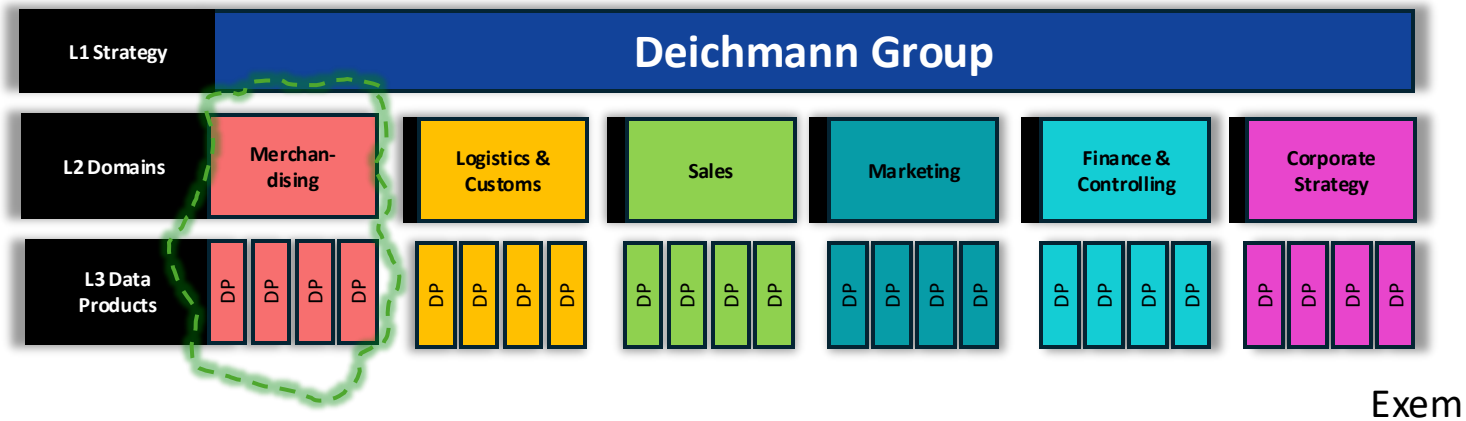
01.

Data Mesh @ Deichmann

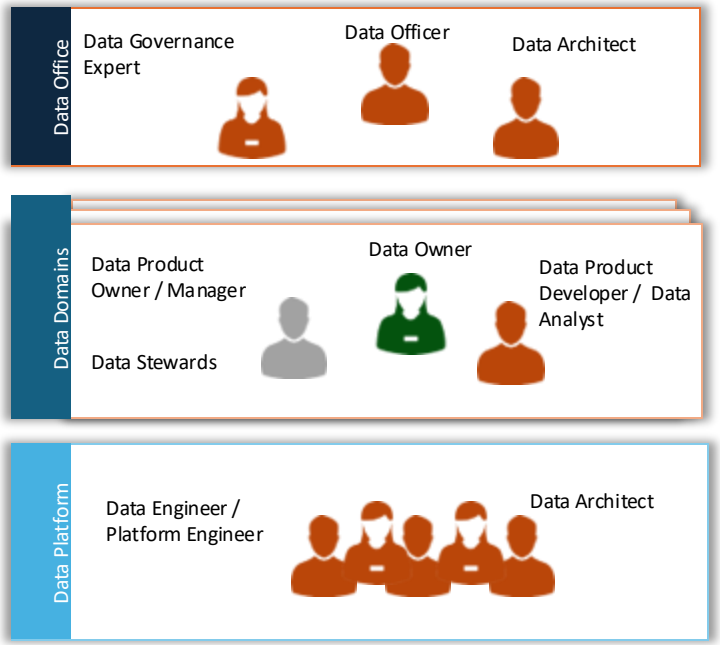
Deichmanns “Road to Data Mesh”

Data Domains* @ Deichmann

Auszug von Domänen in der Deichmann Gruppe, mit ihren jeweiligen Datenprodukten

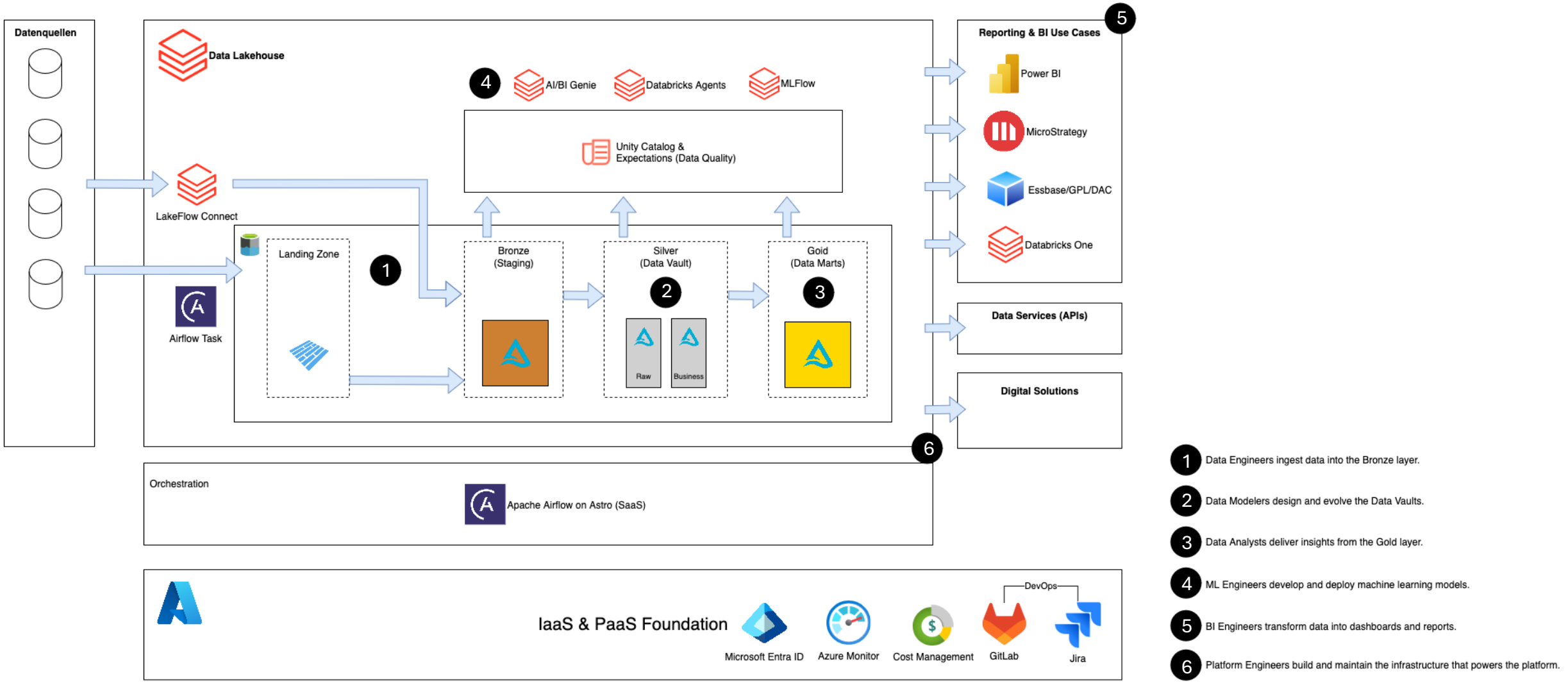


Exemplarische Rollen in einer Domäne



*unvollständig und stark vereinfacht

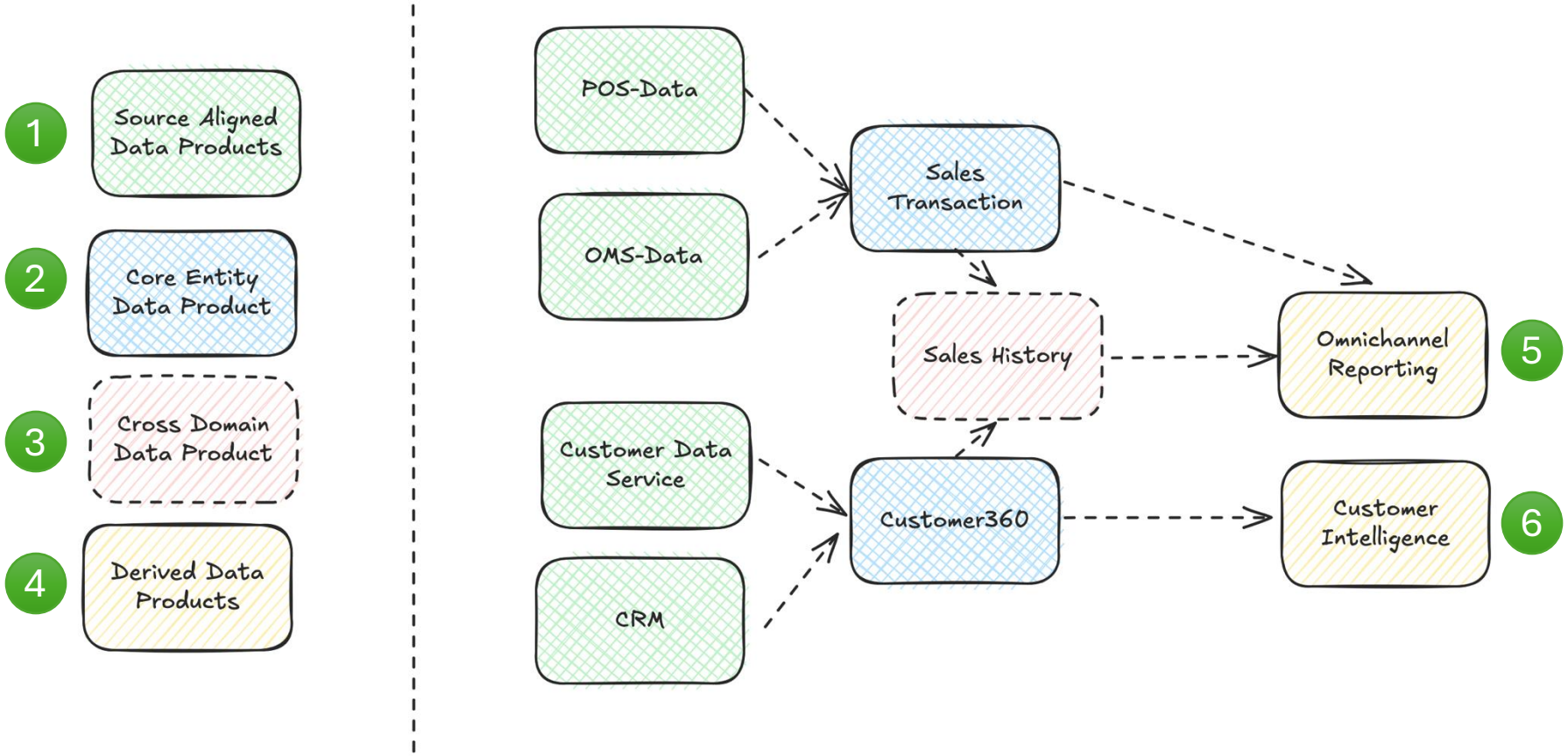
Deichmanns Data Analytics Platform (DAP)



1. Data Engineers ingest data into the Bronze layer.
2. Data Modelers design and evolve the Data Vaults.
3. Data Analysts deliver insights from the Gold layer.
4. ML Engineers develop and deploy machine learning models.
5. BI Engineers transform data into dashboards and reports.
6. Platform Engineers build and maintain the infrastructure that powers the platform.

Data Products & Data Assets

Bei Deichmann verfolgen wir eine Datenproduktstrategie, in der wir die **Datenprodukte inkl. ihrer Data Assets** bündeln und eine **klare Ownership** aufbauen (wollen). Die Basis bietet unser **Data Lakehouse** auf Azure Databricks und der **Medaillon Architecture**.



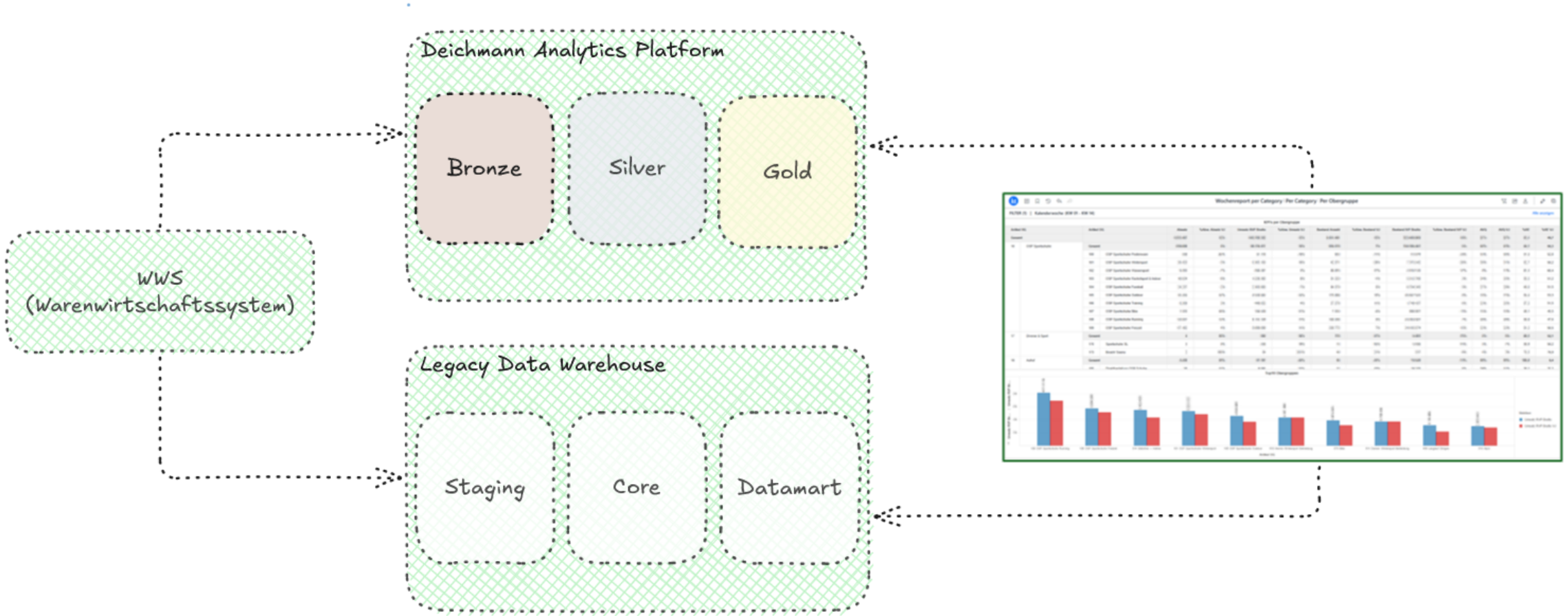
1. Qualitätsgesichertes Datenprodukt, angelehnt an die Datenquelle
2. Core-Datenprodukt mit Data Vault im Silver-Layer
3. Datenprodukt zur Verbindung von 1:n Domänen
4. Kombiniertes Datenprodukt aus Vorstufen (in der Regel nur noch Gold-Layer)
5. Beispiel der Reporting Landschaft für die Omnichannel Geschäftssteuerung
6. Beispiel für Produkte um Customer Segmentation, Churn Forecast etc.

02.

Use Case “DWH Migration“

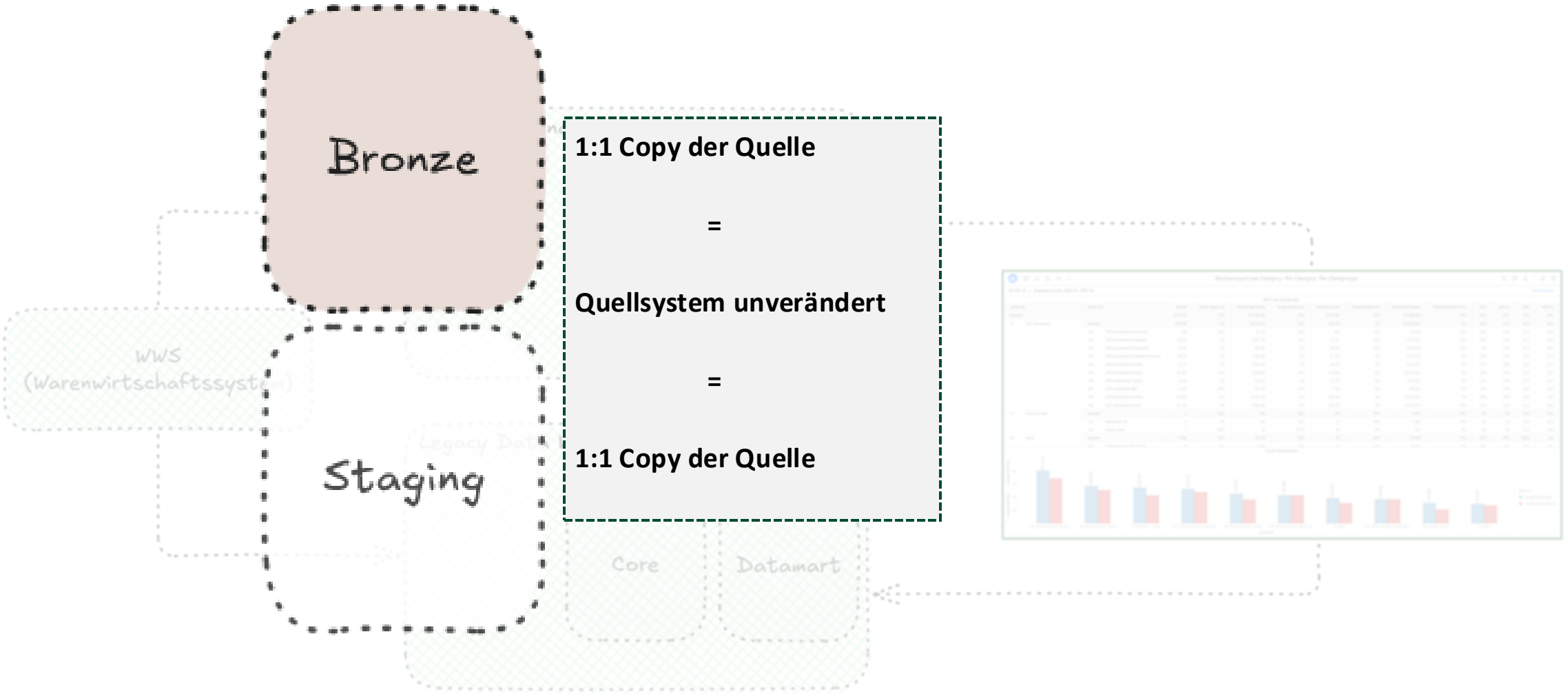
Ausgangslage

Migration der Datenbasis eines Dashboards vom Legacy DWH auf neue Analytics Platform.



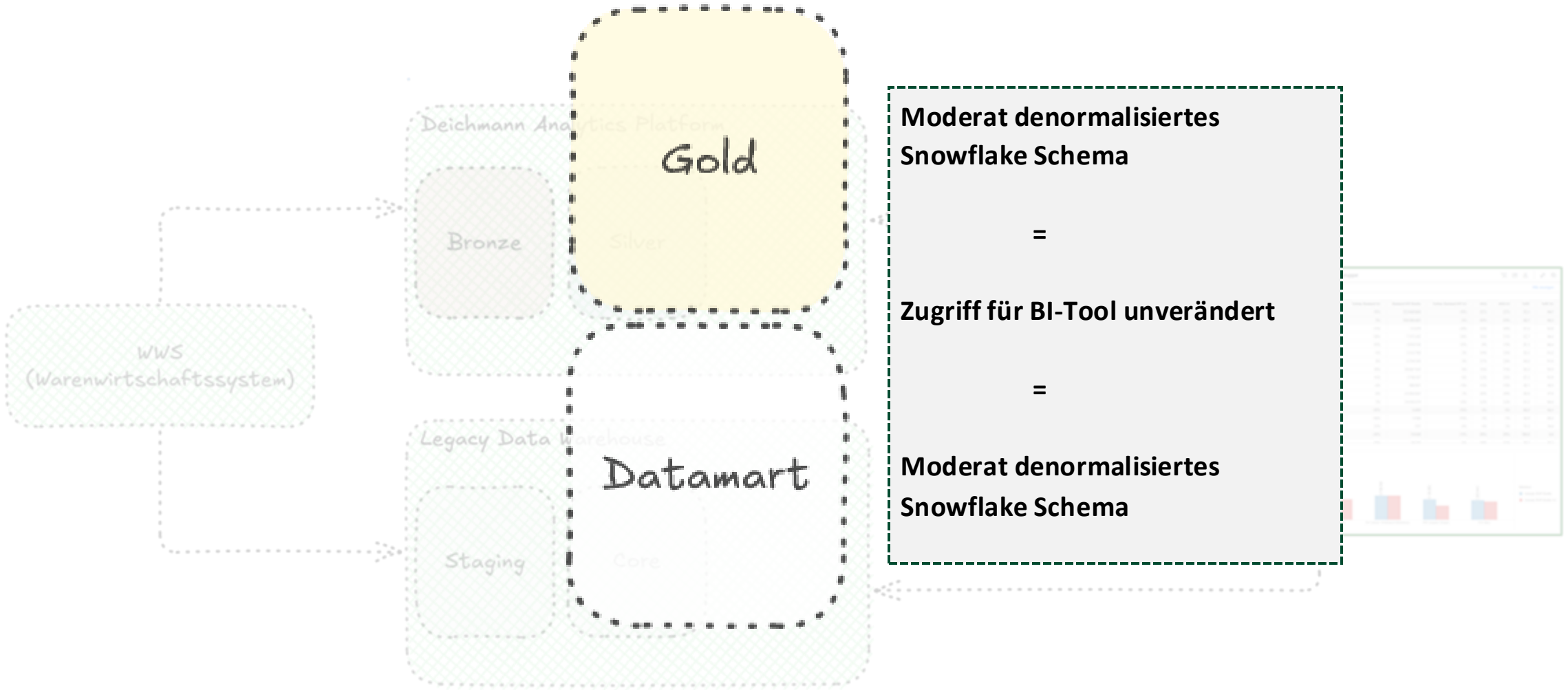
Ausgangslage

Migration der Datenbasis eines Dashboards vom Legacy DWH auf neue Analytics Platform.



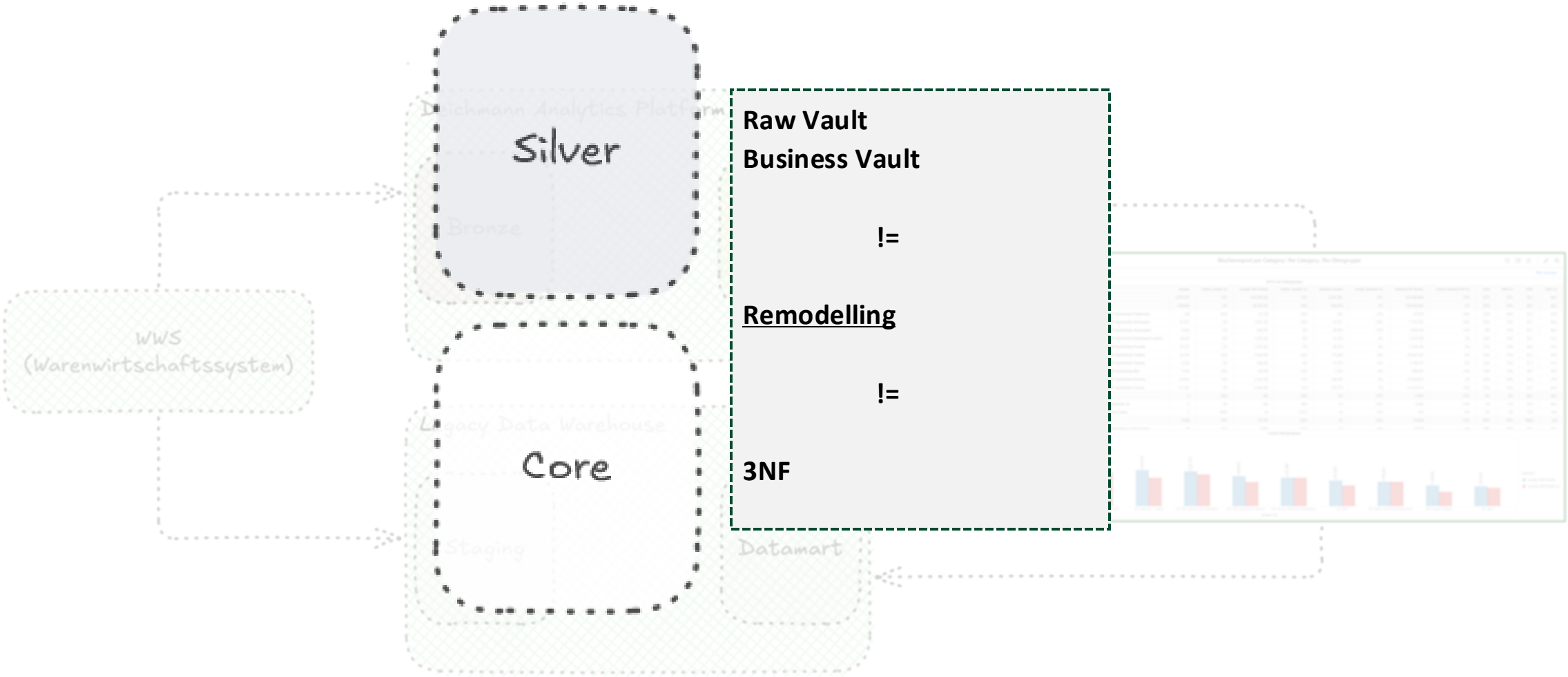
Ausgangslage

Migration der Datenbasis eines Dashboards vom Legacy DWH auf neue Analytics Platform.



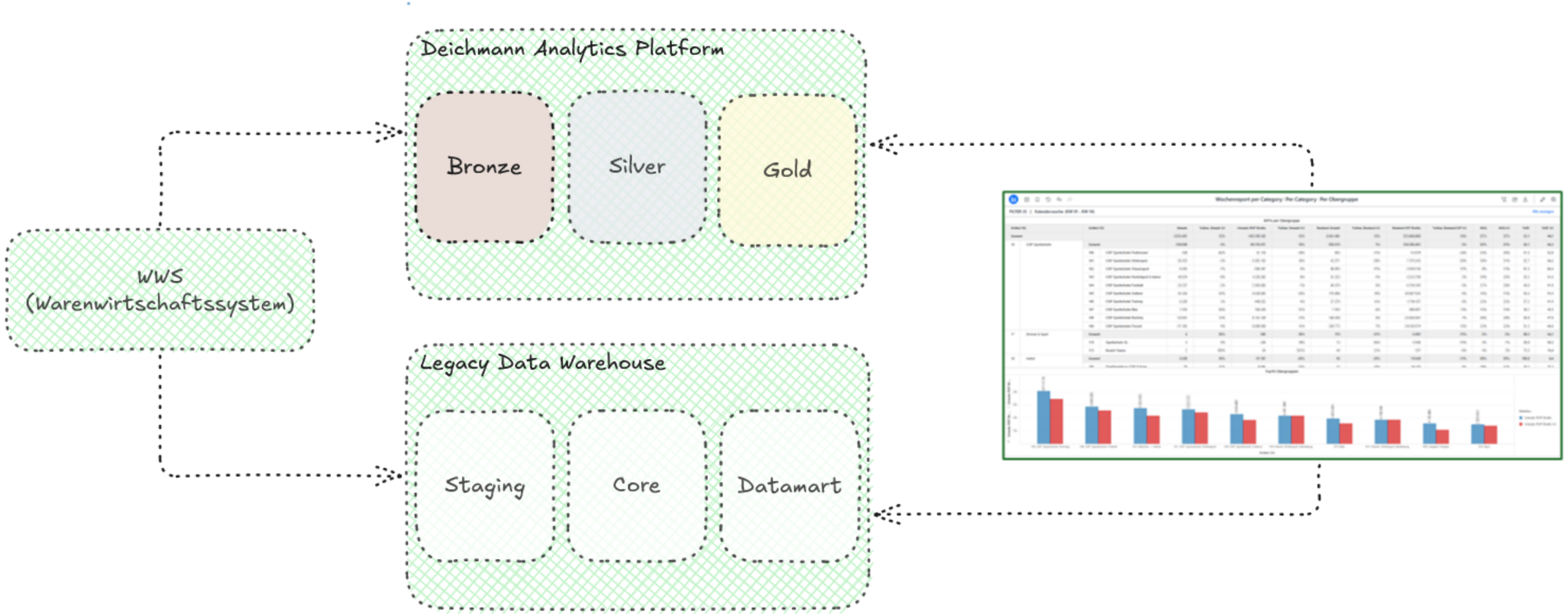
Ausgangslage

Migration der Datenbasis eines Dashboards vom Legacy DWH auf neue Analytics Platform.



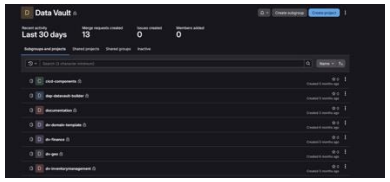
Ausgangslage

Migration der Datenbasis eines Dashboards vom Legacy DWH auf neue Analytics Platform.



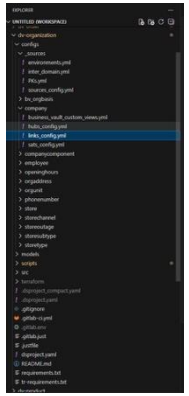
Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.




```yaml
orgein_company_organization:
  source_table: wws_bronze.orgein
  source_type: bronze
  transformations:
    filters:
      - subtyp = 'F'
    renames:
      - [nr_benutzersicht, bk_nr_benutzersicht_company]
      - [STAAT_NR_DEKLARANT, bk_nr_legalentity]
      - [DAT_AKTIV_VON, eff_from]
      - [DAT_AKTIV_BIS, eff_to]
  entity_metadata:
    rec_src: wws.orgein
```

```yaml
company:
  source_name: orgein_company_organization
  business_keys:
    - bk_nr_benutzersicht_company
  table_properties:
    source_system: WWS
    description: Master data for organizational units of type company
```

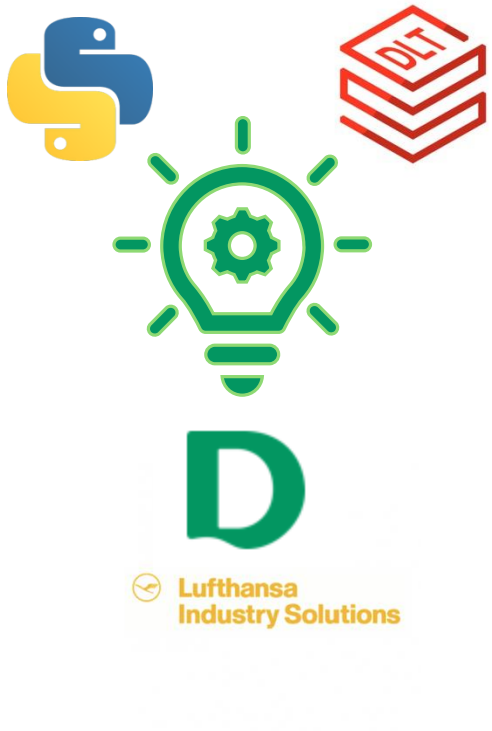


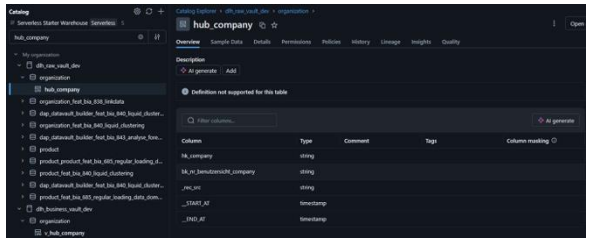
```yaml
company:
  - sat_name: active
    sat_type: effectivity
    table_properties:
      source_system: wws
      description: Effectivity tracking of company records to identify active or valid periods

  - sat_name: reportctrl
    sat_type: standard
    columns:
      - nr_ordnung
      - ref_cois_relevant
    table_properties:
      source_system: wws
      description: Attributes controlling reporting relevance or system inclusion of companies

  - sat_name: details
    sat_type: standard
    columns:
      - nr
      - subtyp
      - ref_typ
```

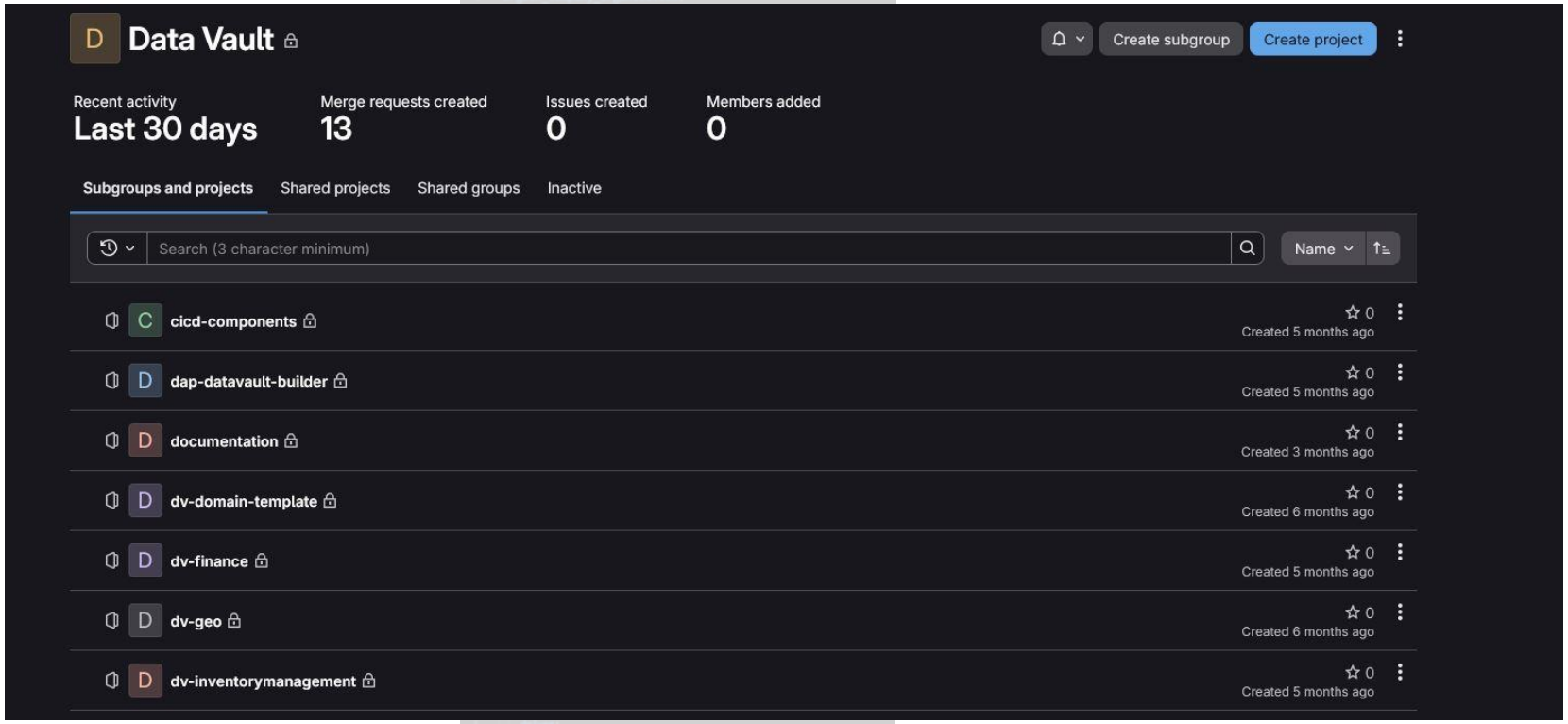
```yaml
company_currency:
  source_name: join_company_currency
  link_type: standard
  hubs:
    driver:
      - company
    secondary:
      - currency
  effectivity_satellite:
    status: active
  table_properties:
    source_system: wws
    description: Link between companies and their main currencies
```





Data Vault Toolkit

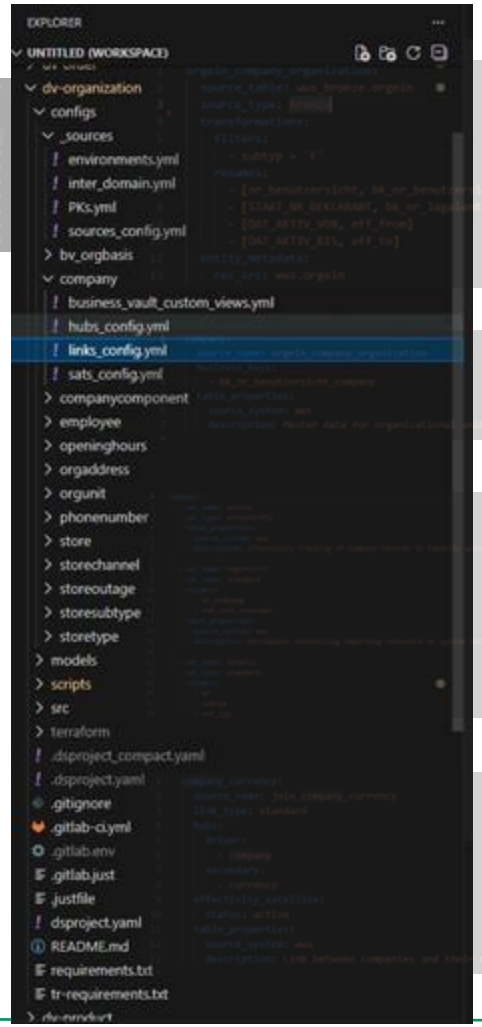
Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.



- **Framework getrennt von Datendefinition**
- **Gitlab Repo je Datenprodukt**
- **Template Repo als Vorlage**

Data Vault Toolkit

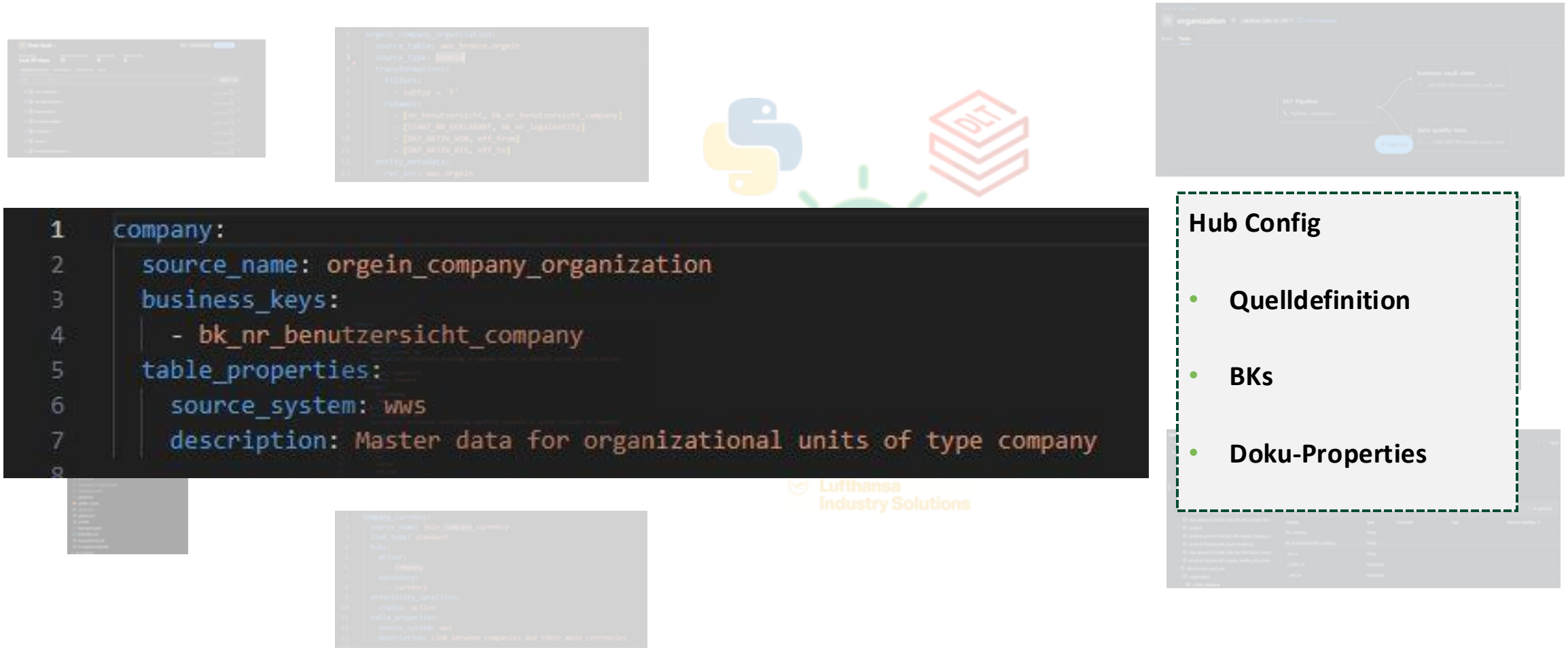
Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.



- **Config Files für Tabellen**
- **Projekt Configs für einfaches Deployment**
- **Mermaid-Models**

Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.



```yaml
company:
  source_name: orgein_company_organization
  business_keys:
    - bk_nr_benutzersicht_company
  table_properties:
    source_system: wws
    description: Master data for organizational units of type company
```

Hub Config

- **Quelldefinition**
- **BKs**
- **Doku-Properties**

Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.

```yaml
orgein_company_organization:
  source_table: wws_bronze.orgein
  source_type: bronze
  transformations:
    filters:
      - subtyp = 'F'
    renames:
      - [nr_benutzersicht, bk_nr_benutzersicht_company]
      - [STAAT_NR_DEKLARANT, bk_nr_legalentity]
      - [DAT_AKTIV_VON, eff_from]
      - [DAT_AKTIV_BIS, eff_to]
  entity_metadata:
    rec_src: wws.orgein
```

Source Config

- **Einfache Transformationen in YAML**
- **Komplexere Definitionen als SQL**
- **Wiederverwendbar**

Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarativ

```yaml
company:
  - sat_name: active
    sat_type: effectivity
    table_properties:
      source_system: wws
      description: Effectivity tracking of company records to identify active or valid periods

  - sat_name: reportctrl
    sat_type: standard
    columns:
      - nr_ordnung
      - ref_cois_relevant
    table_properties:
      source_system: wws
      description: Attributes controlling reporting relevance or system inclusion of companies

  - sat_name: details
    sat_type: standard
    columns:
      - nr
      - subtyp
      - ref_typ
```

Sat Config

- **Spalten können angegeben werden**
- **BKs des Hub müssen in Quelle sein**
- **Abweichende Quellen möglich**

```yaml
company_currency:
  source_name: join_company_currency
  link_type: standard
  hubs:
    driver:
      - company
    secondary:
      - currency
  effectivity_satellite:
    status: active
  table_properties:
    source_system: wws
    description: Link between companies and their main currencies
```

Link Config

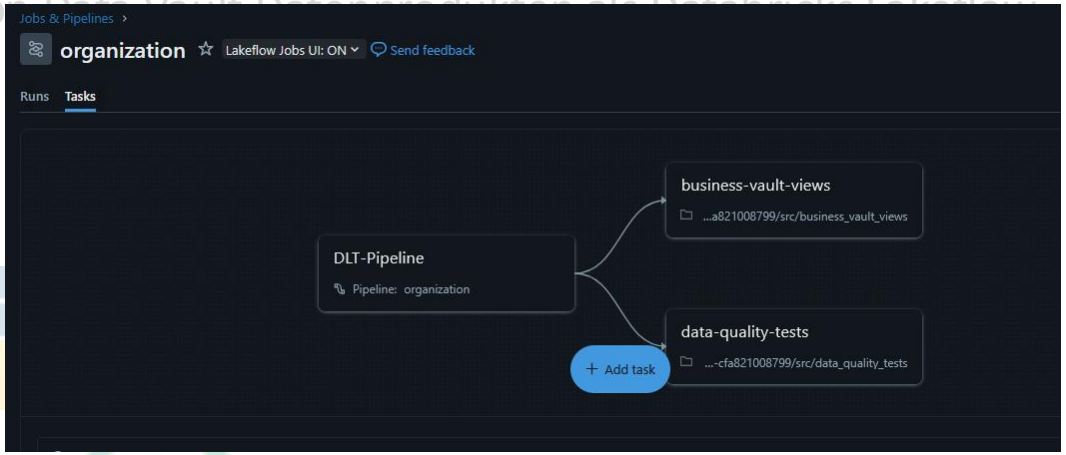
- **BKs aller Hubs müssen in Quelle sein**

Data Vault Toolkit

Entwicklung eines Fra g von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.

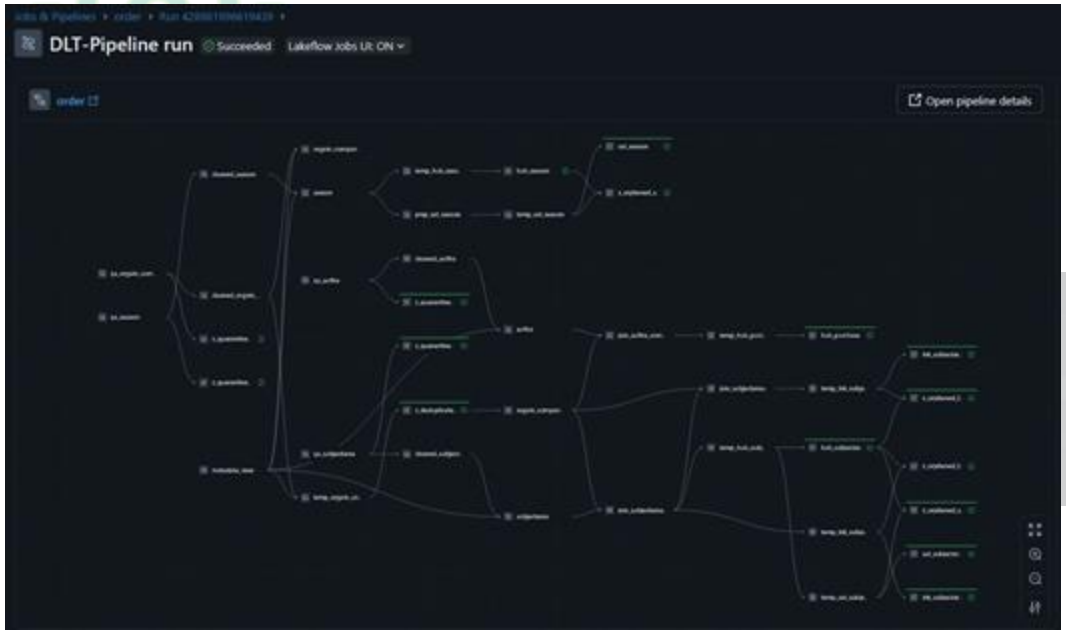
Databricks Job

- DLT-Pipeline
- Business Vault View-Aufbau bei Änderungen
- Data Quality Checks



DLT-Pipeline

- Abhängigkeiten automatisch
- Automatische Increment-Erkennung

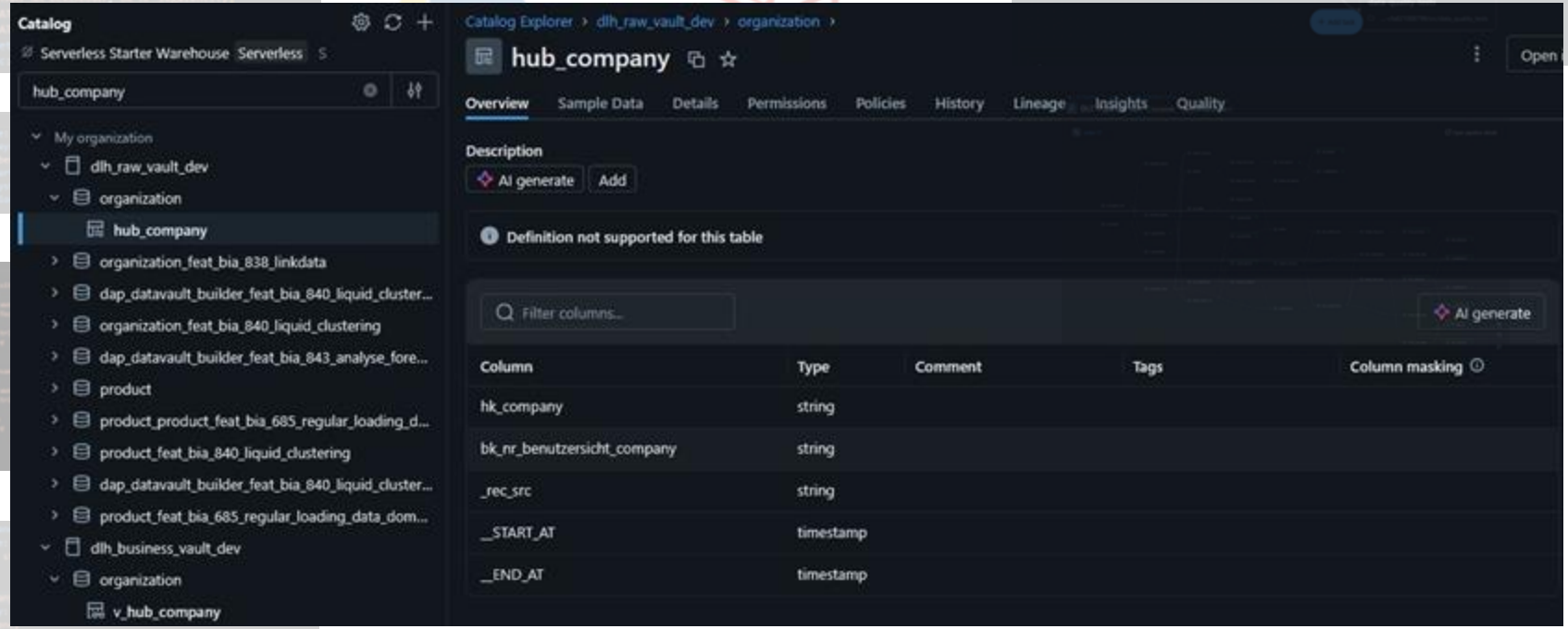


Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.

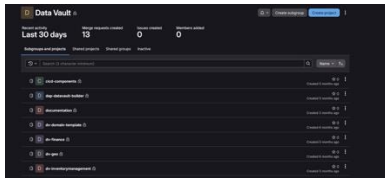
Unity Catalog

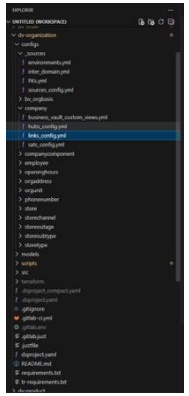
- **Raw Vault / Business Vault auf Katalogebene getrennt**
- **Domäne / Datenprodukt als Schema**



Data Vault Toolkit

Entwicklung eines Frameworks zur Implementierung von Data Vault Datenprodukten als Databricks Lakeflow Declarative Pipelines.





```yaml
orgein_company_organization:
  source_table: wws_bronze.orgein
  source_type: bronze
  transformations:
    filters:
      - subtyp = 'F'
    renames:
      - [nr_benutzersicht, bk_nr_benutzersicht_company]
      - [STAAT_NR_DEKLARANT, bk_nr_legalentity]
      - [DAT_AKTIV_VON, eff_from]
      - [DAT_AKTIV_BIS, eff_to]
  entity_metadata:
    rec_src: wws.orgein
```

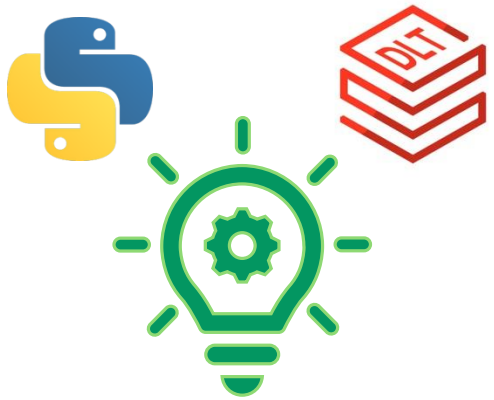
```yaml
company:
  source_name: orgein_company_organization
  business_keys:
    - bk_nr_benutzersicht_company
  table_properties:
    source_system: WWS
    description: Master data for organizational units of type company
```

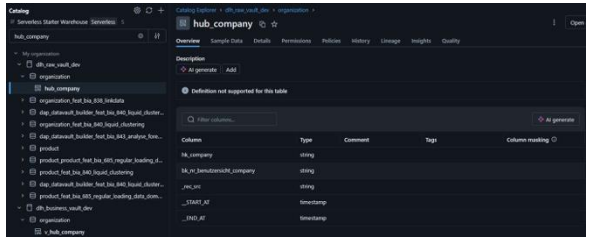
```yaml
company:
  - sat_name: active
    sat_type: effectivity
    table_properties:
      source_system: wws
      description: Effectivity tracking of company records to identify active or valid periods

  - sat_name: reportctrl
    sat_type: standard
    columns:
      - nr_ordnung
      - ref_cois_relevant
    table_properties:
      source_system: wws
      description: Attributes controlling reporting relevance or system inclusion of companies

  - sat_name: details
    sat_type: standard
    columns:
      - nr
      - subtyp
      - ref_typ
```

```yaml
company_currency:
  source_name: join_company_currency
  link_type: standard
  hubs:
    driver:
      - company
    secondary:
      - currency
  effectivity_satellite:
    status: active
  table_properties:
    source_system: wws
    description: Link between companies and their main currencies
```





Challenges DWH Prototyp

Keys

Legacy DWH stark auf Surrogate Keys ausgerichtet.
Business Keys müssen identifiziert werden.

Abgrenzung

Wie schneiden wir die Datendomänen?
Was gehört noch in den Business Vault, was schon in den Serving Layer?

Stammdaten

Keine abgeschlossenen Datenprodukte, Nutzung über Domänengrenzen hinweg

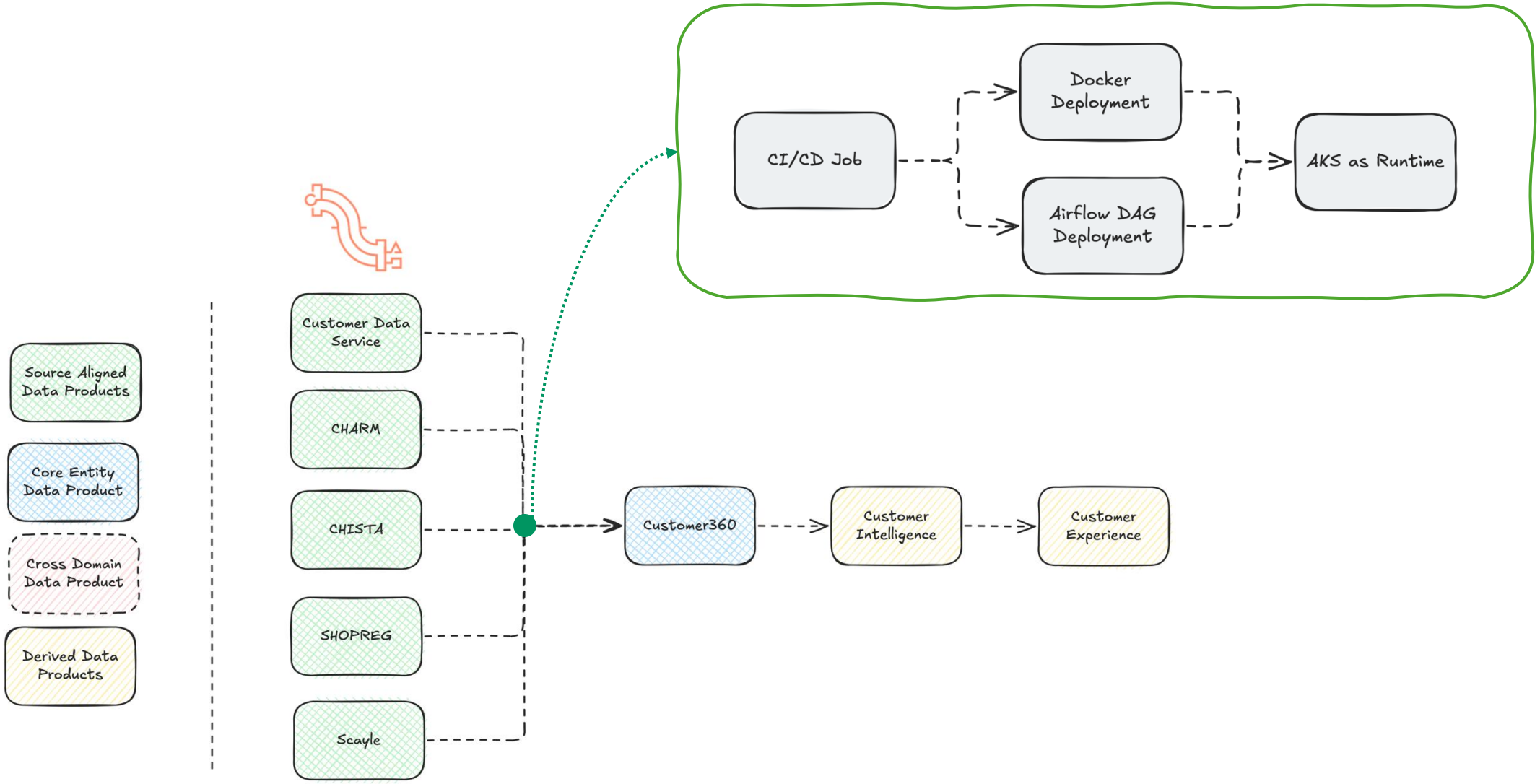
Technische Restriktionen

Nicht alles in DLT umsetzbar. Es mussten Workarounds gefunden und kleinere Abweichungen vom Data Vault Standard in Kauf genommen werden.

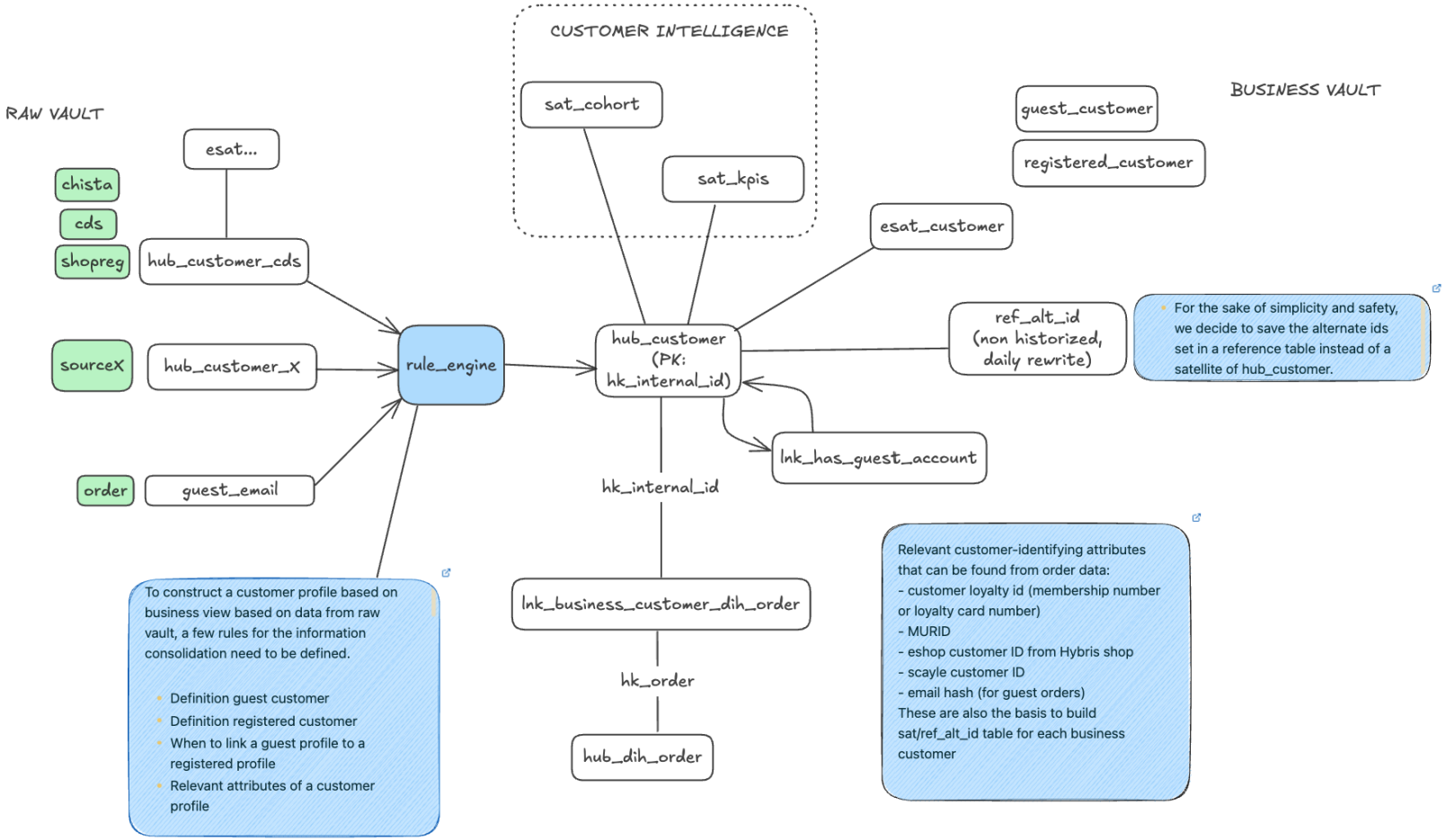
03.

Use Case “Customer Analytics“

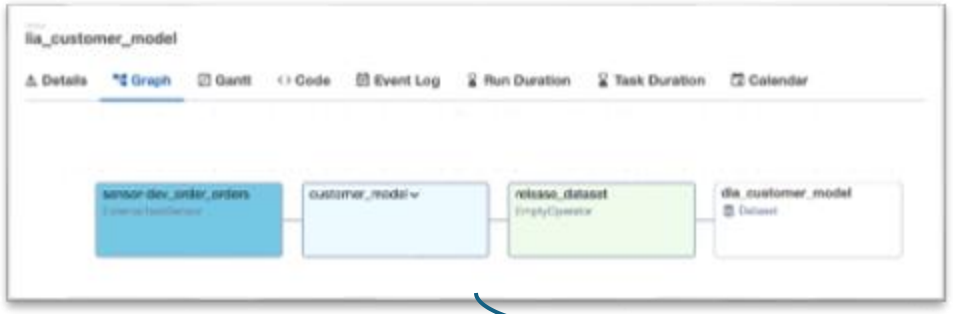
Beispiel von da DATAVAULT4dbt POWERED BY SCALEFREE tomer Analytics Domain



Auszug aus dem Customer Data Model



datavault4dbt & Astronomer Cosmos im DataHub Apache Airflow



Airflow DAG führt dbt Projekt aus und liefert das Data Asset “dia_customer_model“ zur weiteren Orchestrierung aus.

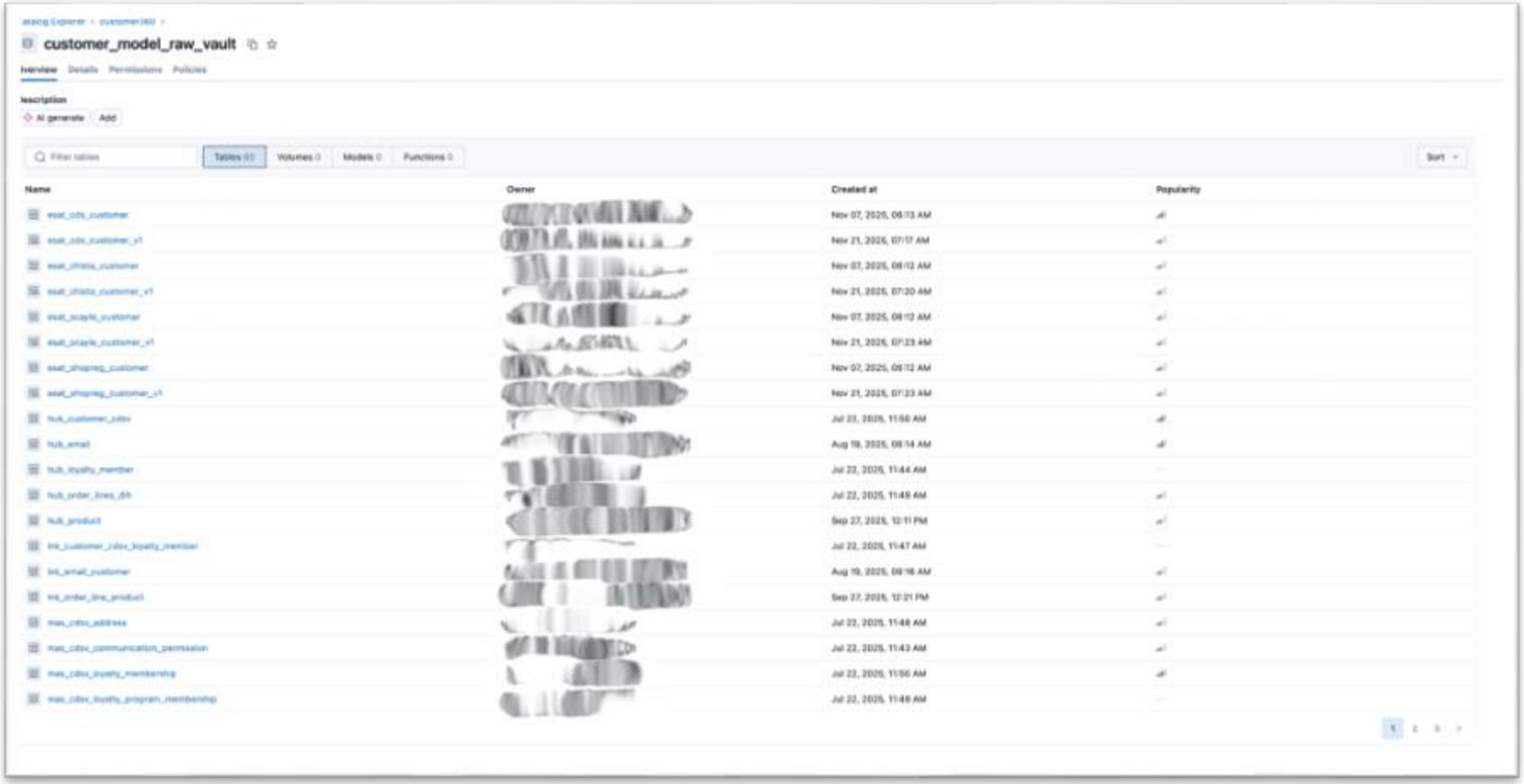
Expandiert: Die Lineage aus dbt im DAG via Cosmos. Ausgeführt mittels deferred Operators auf unserem AKS.

Das (Teil-)Ergebnis – es lebt!

Auszug aus der Raw Vault – der Weg zum goldenen Record in der Business Vault

Disclaimer: **GDPR**

- ✓ Seperater Storage
- ✓ Zugriff nur auditiert
- ✓ Data Masking
- ✓ Löschkonzept implementiert
- ✓ Trennung auf PII-Satelliten



04.

Was haben wir gelernt?

Lessons Learned

1. Gemeinsame BK Definitionen sind der Schlüssel zum Erfolg
2. Automation als Skalierungsfaktor
3. Einheitliche Definitionen und Styleguides (trotz unterschiedlicherAnsätze)
4. Test-Strategie frühzeitig überlegen
5. SCRUM als Treiber

So erreichst du uns:

Sebastian Ennigkeit

sebastian_ennigkeit@deichmann.com

Kevin Haferkamp

kevin_haferkamp@deichmann.com

Wir suchen dich!

WERKSTUDENT (m/w/d)

PRAKTIKANT (m/w/d)

PROFESSIONAL (m/w/d)

Bewirb dich jetzt!

deichmann-karriere.de