DDVUG

Current progress and limitations of GenAI in the field of Data Management

Evaluating GenAI using Willibald

Sebastian Bruns
Hamburg, April 12th 2024

eraneos

We prompted …

“Draw a picture of a Data vault driven by Artificial intelligence”

Your speaker

Sebastian Bruns
Lead Data Engineer
Eraneos Analytics

sebastian.bruns@eraneos.com

The speaker is a **technology enthusiast** and loves to discuss DevOps and CI/CD topics

Sebastian's background is **Computer Science** with focus on **Data Engineering**

- 2007 Oracle
- 2010 SAP Business Warehouse
- 2014 SQL Analysis Services
- 2015 NoSQL Databases
- 2016 "Big Data" / ELT
- 2019 Cloud Engineering

Agenda

The power of Generative AI (GenAI) in automation

How we use Large Language Models (LLMs) to tackle complex automation tasks

GenAI applications in Data Warehouse Automation (DWA)

Taking on the Willibald challenge: Evaluation of LLMs in the realm of the DWA benchmark dataset

Looking Ahead

What we expect from the development in the field of GenAI in 2024 and beyond

Introduction Eraneos

We were experts in Machine Learning long before the rise of GenAI. Now we see the opportunity to solve many problems in a faster and more cost efficient way.

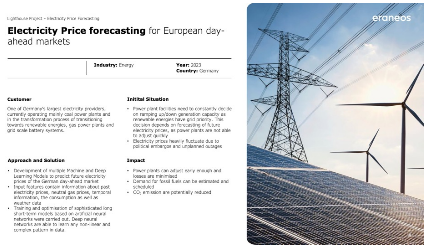
150+
Data & AI Top Experts

500+
Data & AI Projects

150+
Customers in various industries

eraneos

Electricity Price Forecasting
Energy Supplier



Lighthouse Project – Electricity Price Forecasting

Electricity Price forecasting for European day-ahead markets

Industry: Energy
Year: 2023
Country: Germany

Customer

One of Germany's largest electricity providers, currently operating mainly coal power plants and in the transformation process of transitioning towards renewable energies, gas power plants and grid scale battery systems.

Initital Situation

- Power plant facilities need to constantly decide on ramping up/down generation capacity as renewable energies have grid priority. This decision depends on forecasting of future electricity prices, as power plants are not able to adjust quickly
- Electricity prices heavily fluctuate due to political embargos and unplanned outages

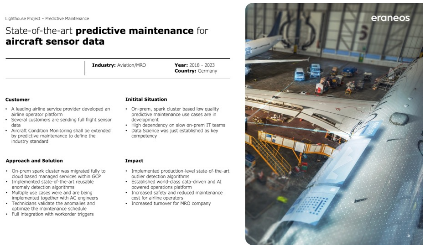
Approach and Solution

- Development of multiple Machine and Deep Learning Models to predict future electricity prices of the German day-ahead market
- Input features contain information about past electricity prices, neutral gas prices, temporal information, the consumption as well as weather data
- Training and optimisation of sophisticated long short-term models based on artificial neural networks were carried out. Deep neural networks are able to learn any non-linear and complex pattern in data.

Impact

- Power plants can adjust early enough and losses are minimised
- Demand for fossil fuels can be estimated and scheduled
- CO_2 emission are potentially reduced

Predictive Maintenance
MRO Service Provider



Lighthouse Project – Predictive Maintenance

State-of-the-art **predictive maintenance** for **aircraft sensor data**

Industry: Aviation/MRO
Year: 2018 - 2023
Country: Germany

Customer

- A leading airline service provider developed an airline operator platform
- Several customers are sending full flight sensor data
- Aircraft Condition Monitoring shall be extended by predictive maintenance to define the industry standard

Initital Situation

- On-prem, spark cluster based low quality predictive maintenance use cases are in development
- High dependency on slow on-prem IT teams
- Data Science was just established as key competency

Approach and Solution

- On-prem spark cluster was migrated fully to cloud based managed services within GCP
- Implemented state-of-the-art reusable anomaly detection algorithms
- Multiple use cases were and are being implemented together with AC engineers
- Technicians validate the anomalies and optimize the maintenance schedule
- Full integration with workorder triggers

Impact

- Implemented production-level state-of-the-art outlier detection algorithms
- Established world-class data-driven and AI powered operations platform
- Increased safety and reduced maintenance cost for airline operators
- Increased turnover for MRO company

AI in Train Disposition
Railway

Lighthouse Project – AI in Train Disposition

Train dispatching is optimized by a digital twin and machine learning

Industry: Railway
Year: 2020 - 2023
Country: Germany

Customer

A railway company currently faces significant challenges related to improving passenger experiences and ensuring punctuality. Recognizing the pivotal role of digital technologies, the company is dedicated to using these to enhance services, boost operational efficiency, and stay competitive in the dynamic railway industry.

Initital Situation

In a bustling urban rail network, numerous real-time manual dispatching decisions are made. The outcomes of these decisions have a clear impact on punctuality and throughput. However, due to the intricate nature of the system, the consequences of these decisions often lack transparency. Inadequate punctuality, in turn, results in reduced customer satisfaction and may lead to contractual penalties.

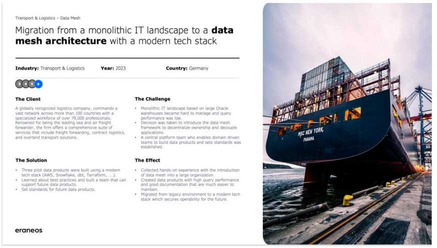
Approach and Solution

- Creating a detailed digital twin by integrating real-world railway network data and utilizing a combination of Reinforcement Learning and other machine learning approaches for data-driven recommendations.
- Providing dispatchers with predictions of future railway network states, potential conflicts, and associated recommendations through an interactive dashboard, which is currently in production.

Impact

- Early conflict detection with immediate alerts for dispatchers.
- Recommendations for dispatching decisions to enhance punctuality and customer satisfaction.
- Data-driven decision support for optimized railway operations.

Legacy DWH Migration
Global Shipping Company



Transport & Logistics – Data Mesh

Migration from a monolithic IT landscape to a **data mesh architecture** with a modern tech stack

Industry: Transport & Logistics **Year:** 2023 **Country:** Germany

3 4 5 9

The Client

A globally recognized logistics company, commands a vast network across more than 100 countries with a specialized workforce of over 79,000 professionals. Renowned for being the leading sea and air freight forwarder, the firm offers a comprehensive suite of services that include freight forwarding, contract logistics, and overland transport solutions.

The Challenge

- Monolithic IT landscape based on large Oracle warehouses became hard to manage and query performance was low.
- Decision was taken to introduce the data mesh framework to decentralize ownership and decouple applications.
- A central platform team who enables domain driven teams to build data products and sets standards was established.

The Solution

- Three pilot data products were built using a modern tech stack (AWS, Snowflake, dbt, Terraform, ...).
- Learned about best practices and built a team that can support future data products.
- Set standards for future data products.

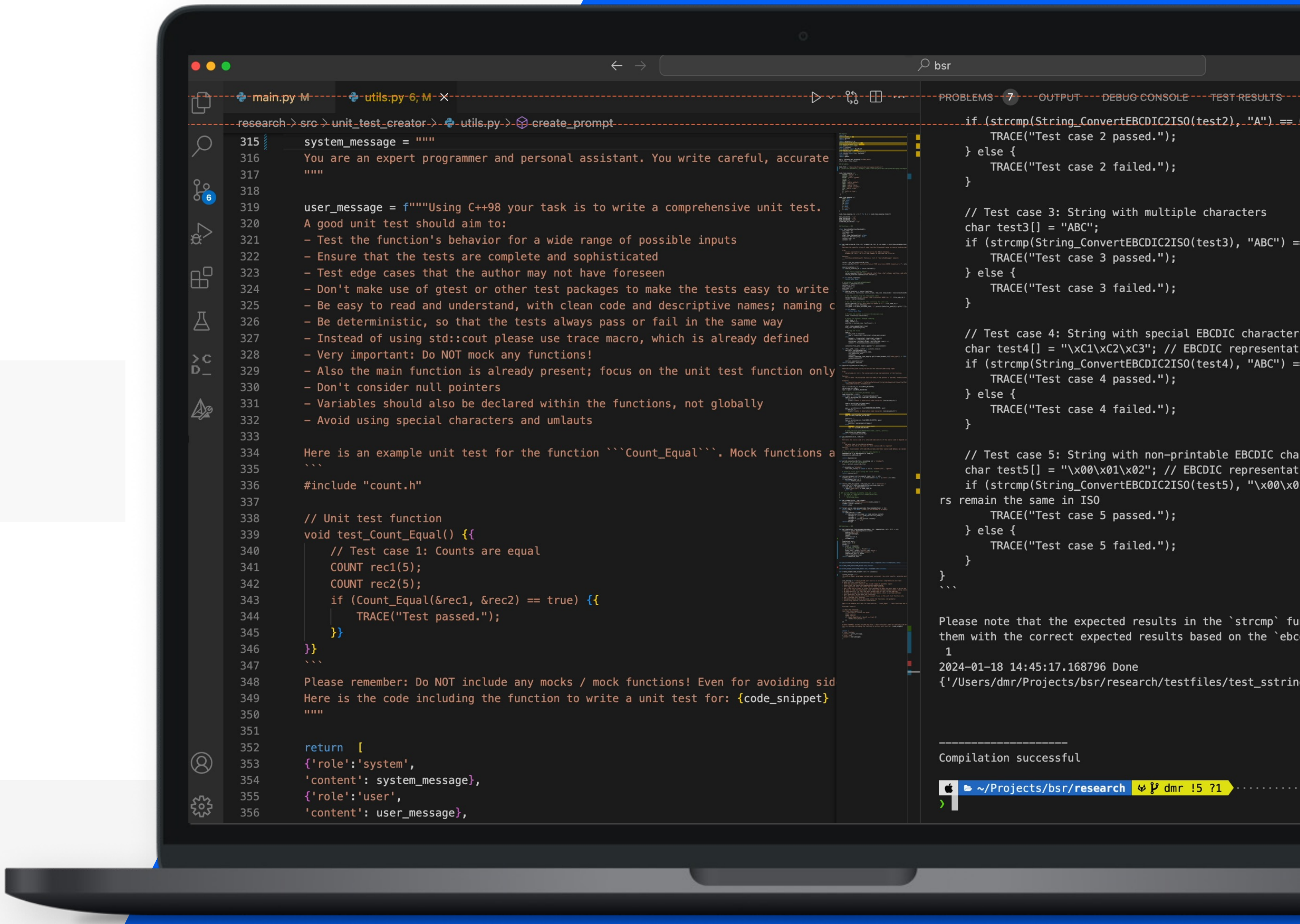
The Effect

- Collected hands-on experience with the introduction of data mesh into a large organization.
- Created data products with high query performance and good documentation that are much easier to maintain.
- Migrated from legacy environment to a modern tech stack which secures operability for the future.

eraneos

The power of Generative AI (GenAI) in automation

How we use Large Language Models (LLMs) to tackle complex automation tasks



```
    system_message = """
    You are an expert programmer and personal assistant. You write careful, accurate
    """

    user_message = f"""Using C++98 your task is to write a comprehensive unit test.
    A good unit test should aim to:
    - Test the function's behavior for a wide range of possible inputs
    - Ensure that the tests are complete and sophisticated
    - Test edge cases that the author may not have foreseen
    - Don't make use of gtest or other test packages to make the tests easy to write
    - Be easy to read and understand, with clean code and descriptive names; naming c
    - Be deterministic, so that the tests always pass or fail in the same way
    - Instead of using std::cout please use trace macro, which is already defined
    - Very important: Do NOT mock any functions!
    - Also the main function is already present; focus on the unit test function only
    - Don't consider null pointers
    - Variables should also be declared within the functions, not globally
    - Avoid using special characters and umlauts

    Here is an example unit test for the function ```Count_Equal```. Mock functions a
    ```
    #include "count.h"

    // Unit test function
    void test_Count_Equal() {{
        // Test case 1: Counts are equal
        COUNT rec1(5);
        COUNT rec2(5);
        if (Count_Equal(&rec1, &rec2) == true) {{
            TRACE("Test passed.");
        }}
    }}
    ```
    Please remember: Do NOT include any mocks / mock functions! Even for avoiding sid
    Here is the code including the function to write a unit test for: {code_snippet}
    """

    return  [
    {'role':'system',
    'content': system_message},
    {'role':'user',
    'content': user_message},
```

```
    if (strcmp(String_ConvertEBCDIC2ISO(test2), "A") ==
        TRACE("Test case 2 passed.");
    } else {
        TRACE("Test case 2 failed.");
    }

    // Test case 3: String with multiple characters
    char test3[] = "ABC";
    if (strcmp(String_ConvertEBCDIC2ISO(test3), "ABC") =
        TRACE("Test case 3 passed.");
    } else {
        TRACE("Test case 3 failed.");
    }

    // Test case 4: String with special EBCDIC character
    char test4[] = "\xC1\xC2\xC3"; // EBCDIC representat
    if (strcmp(String_ConvertEBCDIC2ISO(test4), "ABC") =
        TRACE("Test case 4 passed.");
    } else {
        TRACE("Test case 4 failed.");
    }

    // Test case 5: String with non-printable EBCDIC cha
    char test5[] = "\x00\x01\x02"; // EBCDIC representat
    if (strcmp(String_ConvertEBCDIC2ISO(test5), "\x00\x0
rs remain the same in ISO
        TRACE("Test case 5 passed.");
    } else {
        TRACE("Test case 5 failed.");
    }
}
```

Please note that the expected results in the `strcmp` fu
them with the correct expected results based on the `ebc
 1
2024-01-18 14:45:17.168796 Done
{'/Users/dmr/Projects/bsr/research/testfiles/test_sstrin

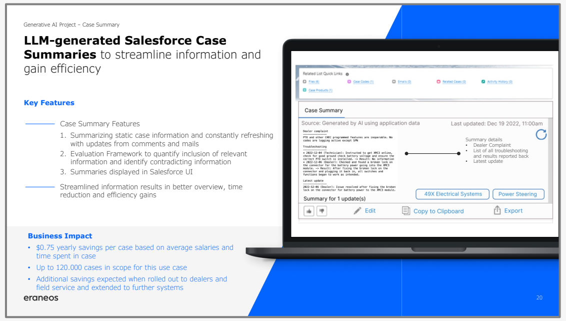
Compilation successful

 ~/Projects/bsr/research dmr !5 ?1
❯
```
```

Eraneos has successfully **leveraged GenAI to automate migration and evaluation tasks** across multiple problem domains and industries

+10 more GenAI projects

LLM-generated Case Summaries
Commercial Vehicles OEM



Generative AI Project – Case Summary

LLM-generated Salesforce Case Summaries to streamline information and gain efficiency

Key Features

- Case Summary Features
  1. Summarizing static case information and constantly refreshing with updates from comments and mails
  2. Evaluation Framework to quantify inclusion of relevant information and identify contradicting information
  3. Summaries displayed in Salesforce UI
- Streamlined information results in better overview, time reduction and efficiency gains

Case Summary

Source: Generated by AI using application data

Last updated: Dec 19 2022, 11:00am

Summary details
- Dealer Complaint
- List of all troubleshooting and results reported back
- Latest update

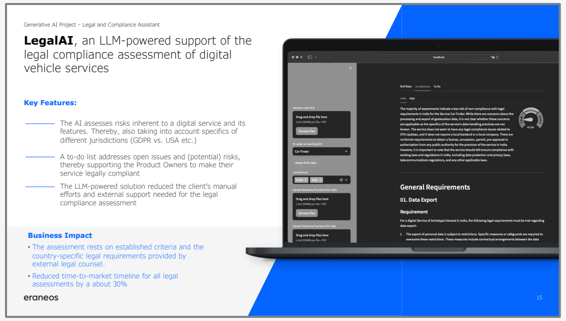
Summary for 1 update(s)

Business Impact
- $0.75 yearly savings per case based on average salaries and time spent in case
- Up to 120.000 cases in scope for this use case
- Additional savings expected when rolled out to dealers and field service and extended to further systems

eraneos 20

Identifying closed tickets that should not have been closed by the service desk

Legal and Compliance Assistant
Automotive



Generative AI Project – Legal and Compliance Assistant

LegalAI, an LLM-powered support of the legal compliance assessment of digital vehicle services

Key Features:

- The AI assesses risks inherent to a digital service and its features. Thereby, also taking into account specifics of different jurisdictions (GDPR vs. USA etc.)
- A to-do list addresses open issues and (potential) risks, thereby supporting the Product Owners to make their service legally compliant
- The LLM-powered solution reduced the client's manual efforts and external support needed for the legal compliance assessment

The majority of assessments indicate a low risk of non-compliance with legal requirements in India for the Service Car Finder. While there are concerns about the processing and export of geolocation data, it is not clear whether these concerns are applicable as the specifics of the service's data handling practices are not known. The service does not seem to have any legal compliance issues related to OTA Updates, and it does not require a local backend or a local company. There are no formal requirements to obtain a license, concession, permit, pre-approval or authorization from any public authority for the provision of the service in India. However, it is important to note that the service should still ensure compliance with existing laws and regulations in India, including data protection and privacy laws, telecommunications regulations, and any other applicable laws.

General Requirements

01. Data Export

Requirement

For a digital Service of Archetype General in India, the following legal requirements must be met regarding data export:

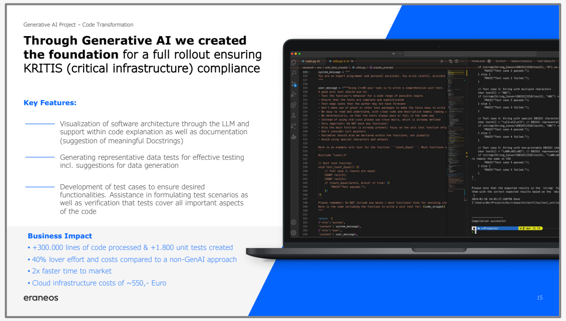
1. The export of personal data is subject to restrictions. Specific measures or safeguards are required to overcome these restrictions. These measures include contractual arrangements between the data

Business Impact
- The assessment rests on established criteria and the country-specific legal requirements provided by external legal counsel.
- Reduced time-to-market timeline for all legal assessments by a about 30%

eraneos 15

Highlight legal regulations for global rollout of products

Code Transformation
Public Sector



Generative AI Project – Code Transformation

Through Generative AI we created the foundation for a full rollout ensuring KRITIS (critical infrastructure) compliance

Key Features:

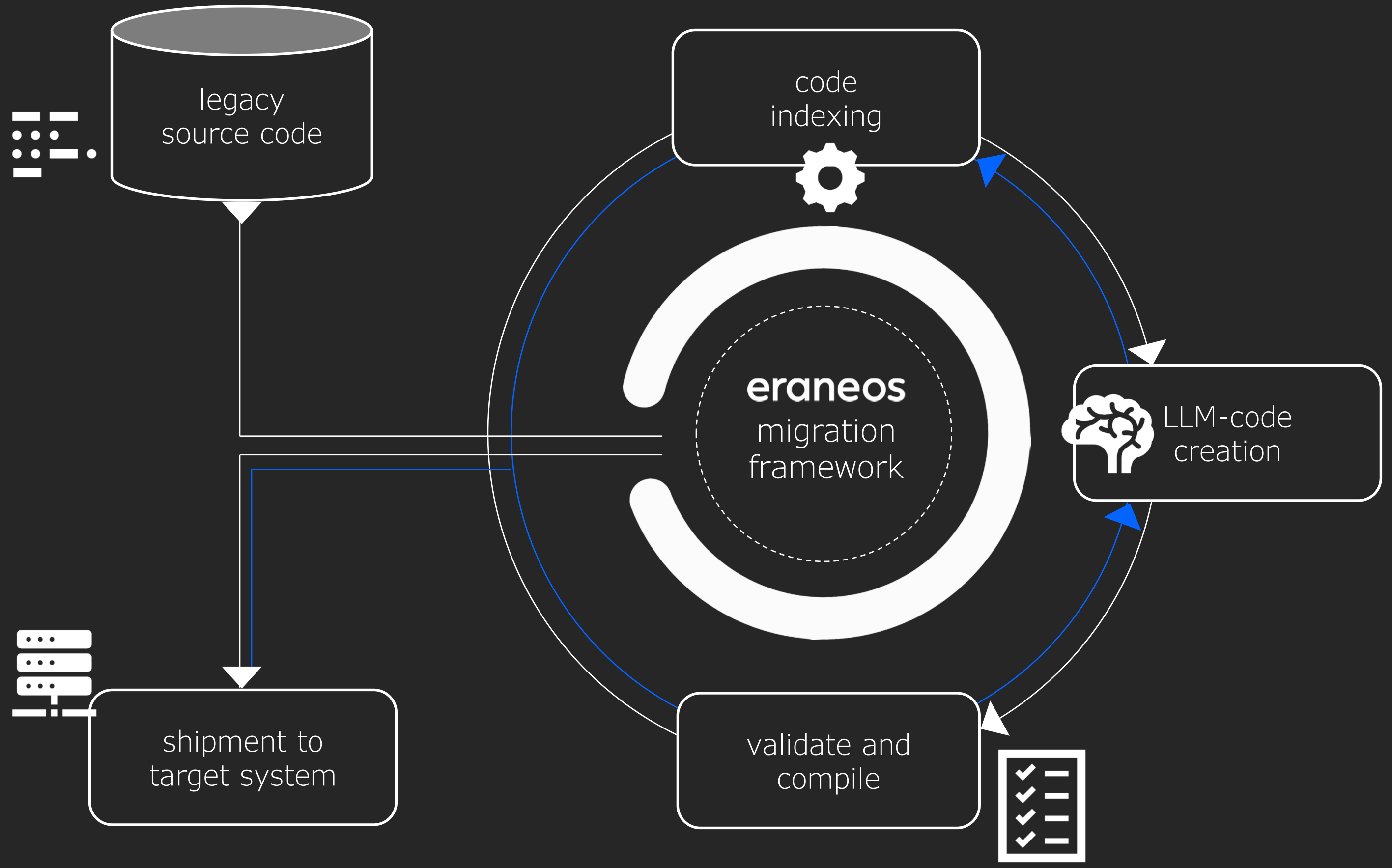
- Visualization of software architecture through the LLM and support within code explanation as well as documentation (suggestion of meaningful Docstrings)
- Generating representative data tests for effective testing incl. suggestions for data generation
- Development of test cases to ensure desired functionalities. Assistance in formulating test scenarios as well as verification that tests cover all important aspects of the code

Business Impact
- +300.000 lines of code processed & +1.800 unit tests created
- 40% lover effort and costs compared to a non-GenAI approach
- 2x faster time to market
- Cloud infrastructure costs of ~550,- Euro

eraneos 15

We have auto-created 1800 Unit Tests on a legacy C++ codebase with >300 K lines of code

Eraneos Code Migration Framework as the core building block for a scalable, reliable and cost-efficient automation around LLMs

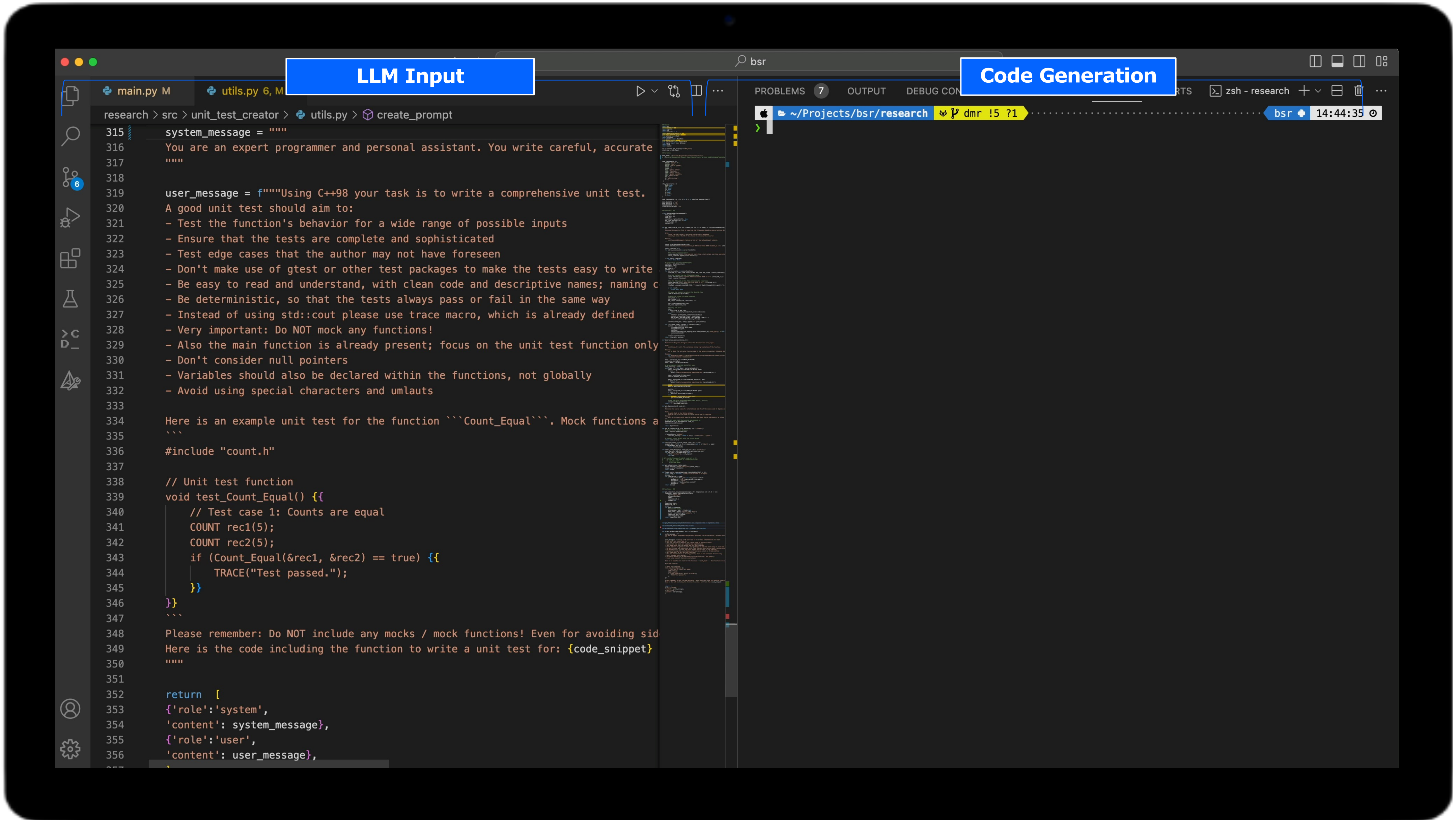


1800+
Unit Test generated

550€
GenAI Bill

80%
Cost & Time Saving

The Power of GenAI – Code Migration Framework Screencast



```python
    system_message = """
    You are an expert programmer and personal assistant. You write careful, accurate
    """

    user_message = f"""Using C++98 your task is to write a comprehensive unit test.
    A good unit test should aim to:
    - Test the function's behavior for a wide range of possible inputs
    - Ensure that the tests are complete and sophisticated
    - Test edge cases that the author may not have foreseen
    - Don't make use of gtest or other test packages to make the tests easy to write
    - Be easy to read and understand, with clean code and descriptive names; naming c
    - Be deterministic, so that the tests always pass or fail in the same way
    - Instead of using std::cout please use trace macro, which is already defined
    - Very important: Do NOT mock any functions!
    - Also the main function is already present; focus on the unit test function only
    - Don't consider null pointers
    - Variables should also be declared within the functions, not globally
    - Avoid using special characters and umlauts

    Here is an example unit test for the function ```Count_Equal```. Mock functions a
    ```
    #include "count.h"

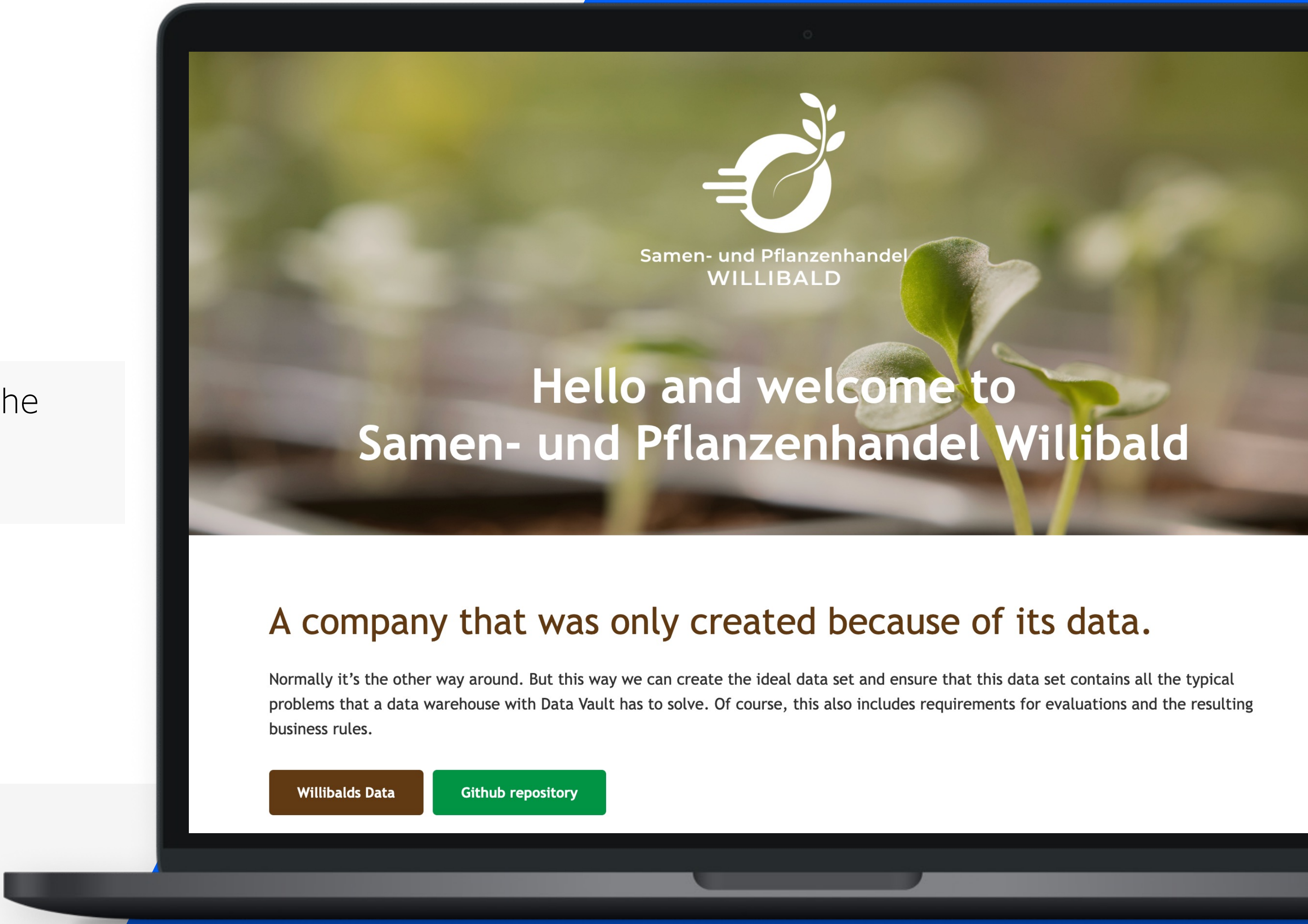
    // Unit test function
    void test_Count_Equal() {{
        // Test case 1: Counts are equal
        COUNT rec1(5);
        COUNT rec2(5);
        if (Count_Equal(&rec1, &rec2) == true) {{
            TRACE("Test passed.");
        }}
    }}
    ```
    Please remember: Do NOT include any mocks / mock functions! Even for avoiding sid
    Here is the code including the function to write a unit test for: {code_snippet}
    """

    return  [
    {'role':'system',
    'content': system_message},
    {'role':'user',
    'content': user_message},
```

```
 ~/Projects/bsr/research  dmr !5 ?1 ································ bsr  14:44:35
❯
```

GenAI applications in Data Warehouse Automation (DWA)

Taking on the Willibald challenge: Evaluation of LLMs in the realm of the DWA benchmark dataset



Hello and welcome to Samen- und Pflanzenhandel Willibald

A company that was only created because of its data.

Normally it’s the other way around. But this way we can create the ideal data set and ensure that this data set contains all the typical problems that a data warehouse with Data Vault has to solve. Of course, this also includes requirements for evaluations and the resulting business rules.

Applications in DWA

The **strength of general-purpose LLMs** is their knowledge and universal applicability to a wide variety of problem statements

Explain & Understand

- Write comments and documentation
- **DEMO** Create field mappings
- **DEMO** Explain the content data and scripts
- Outline data lineage in legacy systems

Generate & Propose

- **DEMO** Create data model from raw data
- Generation of test data (synthetic data)
- **DEMO** Create data model from an ERD
- Recommend refactorings (indexes and contraints and performance improvements)

Assess & Monitor

- Evaluate the functionality of the data model (potentially from business analysts)
- Assess the quality of the data in the database
- Automatically detect anomalies and fraud

Optimize & Migration

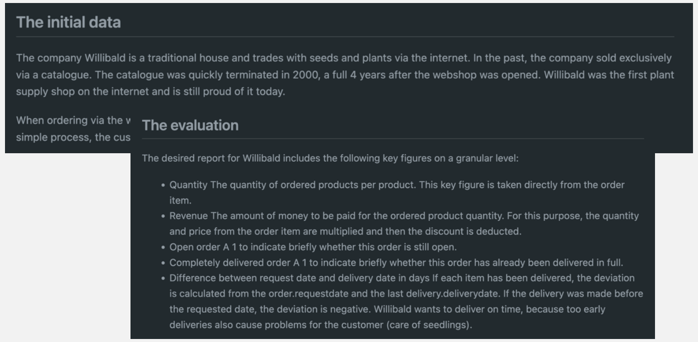
- **DEMO** Migrate the data modeling
- Write data migrate scripts
- Modernize the code base
- Resolve dependencies between (e.g. normalized stacked views)

Creating a basic setup using the markup file and **deducting information about the business context** to build an **initial data vault model**

TASK → PROMPT → RESULT

TASK

Using the [**readme en.md**] markup file as initial starting point



The initial data

The company Willibald is a traditional house and trades with seeds and plants via the internet. In the past, the company sold exclusively via a catalogue. The catalogue was quickly terminated in 2000, a full 4 years after the webshop was opened. Willibald was the first plant supply shop on the internet and is still proud of it today.

When ordering via the w
simple process, the cus

The evaluation

The desired report for Willibald includes the following key figures on a granular level:

- Quantity The quantity of ordered products per product. This key figure is taken directly from the order item.
- Revenue The amount of money to be paid for the ordered product quantity. For this purpose, the quantity and price from the order item are multiplied and then the discount is deducted.
- Open order A 1 to indicate briefly whether this order is still open.
- Completely delivered order A 1 to indicate briefly whether this order has already been delivered in full.
- Difference between request date and delivery date in days If each item has been delivered, the deviation is calculated from the order.requestdate and the last delivery.deliverydate. If the delivery was made before the requested date, the deviation is negative. Willibald wants to deliver on time, because too early deliveries also cause problems for the customer (care of seedlings).

➕ **customer.csv**

Text Input: If we were having larger text input (e.g. documents) we would exceed the token limit at some point and go for RAGs

PROMPT

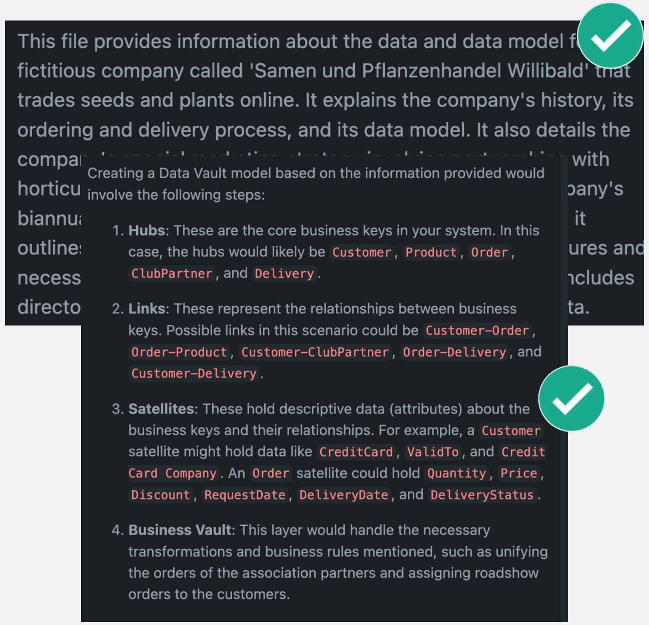
We prompted …

"Give me in brief explanation what Willibald is about"

"Create a Data Vault model reflecting Willibalds business processes"

"What challenges do you see modeling the Willibald data?"
"What customers have no phone number?"

RESULT



This file provides information about the data and data model f
fictitious company called 'Samen und Pflanzenhandel Willibald' that
trades seeds and plants online. It explains the company's history, its
ordering and delivery process, and its data model. It also details the
compar … with
horticu … pany's
biannu … it
outline … ures and
necess … ncludes
directo … ta.

Creating a Data Vault model based on the information provided would involve the following steps:

1. **Hubs**: These are the core business keys in your system. In this case, the hubs would likely be `Customer`, `Product`, `Order`, `ClubPartner`, and `Delivery`.
2. **Links**: These represent the relationships between business keys. Possible links in this scenario could be `Customer-Order`, `Order-Product`, `Customer-ClubPartner`, `Order-Delivery`, and `Customer-Delivery`.
3. **Satellites**: These hold descriptive data (attributes) about the business keys and their relationships. For example, a `Customer` satellite might hold data like `CreditCard`, `ValidTo`, and `Credit Card Company`. An `Order` satellite could hold `Quantity`, `Price`, `Discount`, `RequestDate`, `DeliveryDate`, and `DeliveryStatus`.
4. **Business Vault**: This layer would handle the necessary transformations and business rules mentioned, such as unifying the orders of the association partners and assigning roadshow orders to the customers.

✓

"Incomplete Data" and "Data Quality: There are some fields that appear to have missing"

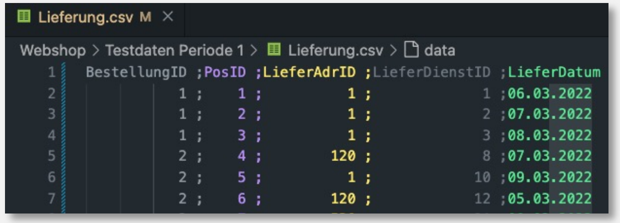
✓

We want to know **how knowledgeable general-purpose LLMs are about modeling techniques** and how they implement them

TASK → PROMPT → RESULT

TASK

The [Lieferung] Data from Willibald's "Webshop Testdata 1" contains references to other tables (e.g. [Bestellung]) as well as actual data (e.g. [LieferDatum)]



```
BestellungID ;PosID ;LieferAdrID ;LieferDienstID ;LieferDatum
           1 ;    1 ;          1 ;             1 ;06.03.2022
           1 ;    2 ;          1 ;             2 ;07.03.2022
           1 ;    3 ;          1 ;             3 ;08.03.2022
           2 ;    4 ;        120 ;             8 ;07.03.2022
           2 ;    5 ;          1 ;            10 ;09.03.2022
           2 ;    6 ;        120 ;            12 ;05.03.2022
```

This is new: LLMs understands the *meaning* of the data fields and model them accordingly.

Non-deterministic: Column names are often, but not always deducted correctly.

PROMPT

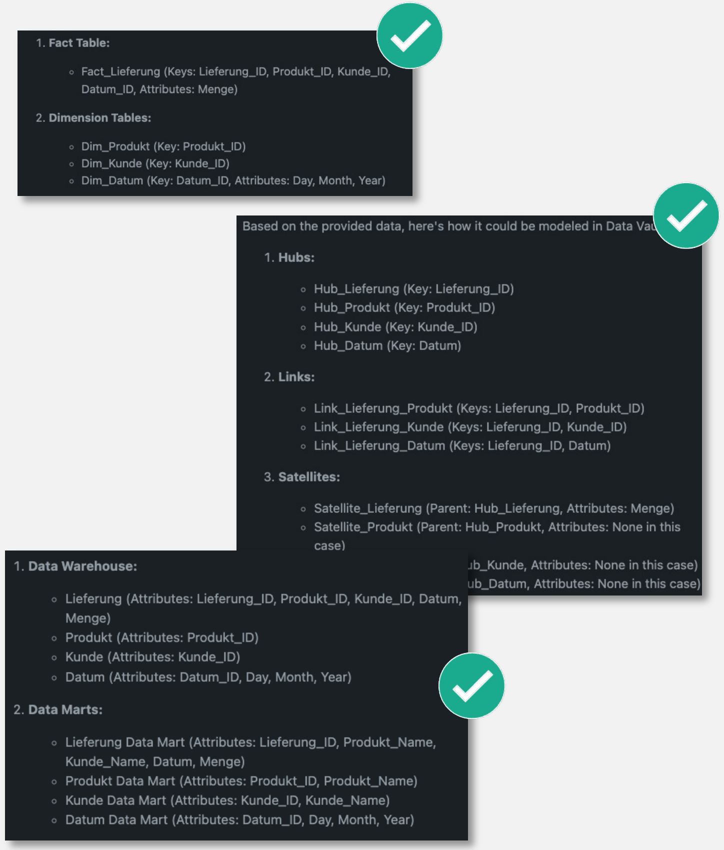
We prompted …

"Model this dataset as **star schema** with fact and dimension tables."

"Model this dataset as **Data Vault** with hubs, links and satellites."

"Model this dataset as **3rd Normal Form**."

RESULT



1. **Fact Table:**
   - Fact_Lieferung (Keys: Lieferung_ID, Produkt_ID, Kunde_ID, Datum_ID, Attributes: Menge)
2. **Dimension Tables:**
   - Dim_Produkt (Key: Produkt_ID)
   - Dim_Kunde (Key: Kunde_ID)
   - Dim_Datum (Key: Datum_ID, Attributes: Day, Month, Year)

Based on the provided data, here's how it could be modeled in Data Va

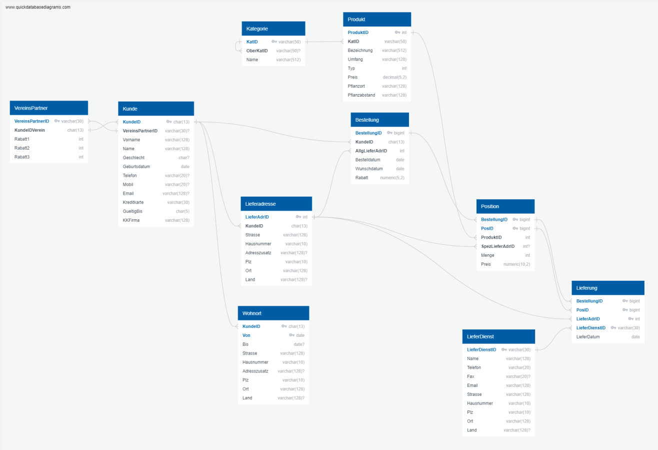
1. **Hubs:**
   - Hub_Lieferung (Key: Lieferung_ID)
   - Hub_Produkt (Key: Produkt_ID)
   - Hub_Kunde (Key: Kunde_ID)
   - Hub_Datum (Key: Datum)
2. **Links:**
   - Link_Lieferung_Produkt (Keys: Lieferung_ID, Produkt_ID)
   - Link_Lieferung_Kunde (Keys: Lieferung_ID, Kunde_ID)
   - Link_Lieferung_Datum (Keys: Lieferung_ID, Datum)
3. **Satellites:**
   - Satellite_Lieferung (Parent: Hub_Lieferung, Attributes: Menge)
   - Satellite_Produkt (Parent: Hub_Produkt, Attributes: None in this case)
   - ub_Kunde, Attributes: None in this case)
   - ub_Datum, Attributes: None in this case)

1. **Data Warehouse:**
   - Lieferung (Attributes: Lieferung_ID, Produkt_ID, Kunde_ID, Datum, Menge)
   - Produkt (Attributes: Produkt_ID)
   - Kunde (Attributes: Kunde_ID)
   - Datum (Attributes: Datum_ID, Day, Month, Year)
2. **Data Marts:**
   - Lieferung Data Mart (Attributes: Lieferung_ID, Produkt_Name, Kunde_Name, Datum, Menge)
   - Produkt Data Mart (Attributes: Produkt_ID, Produkt_Name)
   - Kunde Data Mart (Attributes: Kunde_ID, Kunde_Name)
   - Datum Data Mart (Attributes: Datum_ID, Day, Month, Year)

We have been provided the **source table UML diagram** and wish to create the data vault model including build scripts and field mappings

TASK → PROMPT → RESULT

We provide the LLM the file [webshop-willibald.png] without providing any more information than the prompt



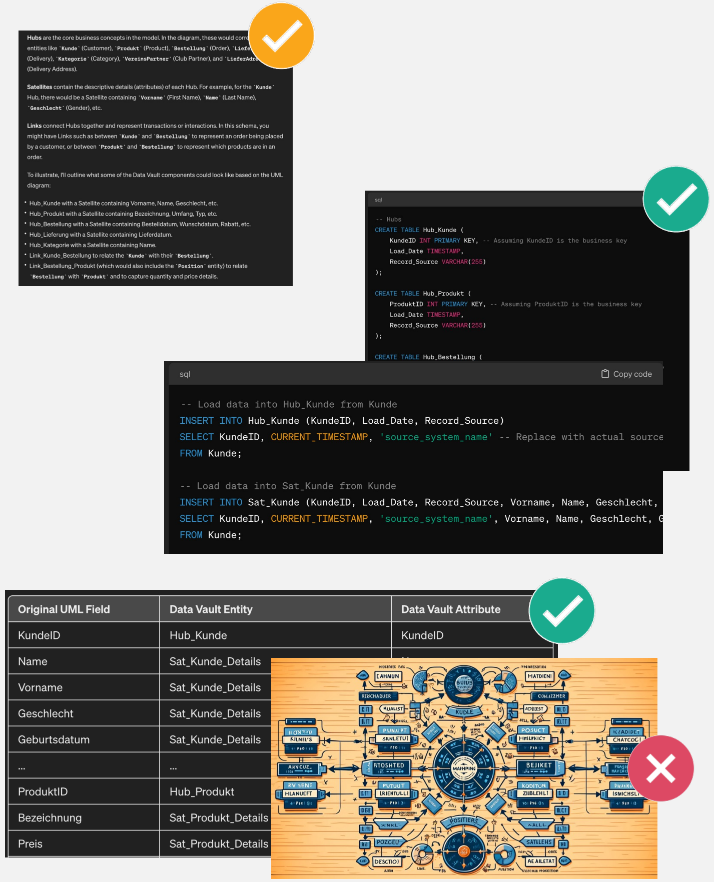
Only two years back, this kind of image recognition task would have taken us a full project!

We prompted …

"Create a data vault model from this UML diagram"

"Create the SQL statements that generate the data vault"

"Draw me a mapping of the fields between the source and the target model"



Hubs are the core business concepts in the model. In the diagram, these would corre entities like `Kunde` (Customer), `Produkt` (Product), `Bestellung` (Order), `Liefe (Delivery), `Kategorie` (Category), `VereinsPartner` (Club Partner), and `LieferAdre (Delivery Address).

Satellites contain the descriptive details (attributes) of each Hub. For example, for the `Kunde` Hub, there would be a Satellite containing `Vorname` (First Name), `Name` (Last Name), `Geschlecht` (Gender), etc.

Links connect Hubs together and represent transactions or interactions. In this schema, you might have Links such as between `Kunde` and `Bestellung` to represent an order being placed by a customer, or between `Produkt` and `Bestellung` to represent which products are in an order.

To illustrate, I'll outline what some of the Data Vault components could look like based on the UML diagram:

- Hub_Kunde with a Satellite containing Vorname, Name, Geschlecht, etc.
- Hub_Produkt with a Satellite containing Bezeichnung, Umfang, Typ, etc.
- Hub_Bestellung with a Satellite containing Bestelldatum, Wunschdatum, Rabatt, etc.
- Hub_Lieferung with a Satellite containing Lieferdatum.
- Hub_Kategorie with a Satellite containing Name.
- Link_Kunde_Bestellung to relate the `Kunde` with their `Bestellung`.
- Link_Bestellung_Produkt (which would also include the `Position` entity) to relate `Bestellung` with `Produkt` and to capture quantity and price details.

sql

```sql
-- Hubs
CREATE TABLE Hub_Kunde (
    KundeID INT PRIMARY KEY, -- Assuming KundeID is the business key
    Load_Date TIMESTAMP,
    Record_Source VARCHAR(255)
);

CREATE TABLE Hub_Produkt (
    ProduktID INT PRIMARY KEY, -- Assuming ProduktID is the business key
    Load_Date TIMESTAMP,
    Record_Source VARCHAR(255)
);

CREATE TABLE Hub_Bestellung (
```

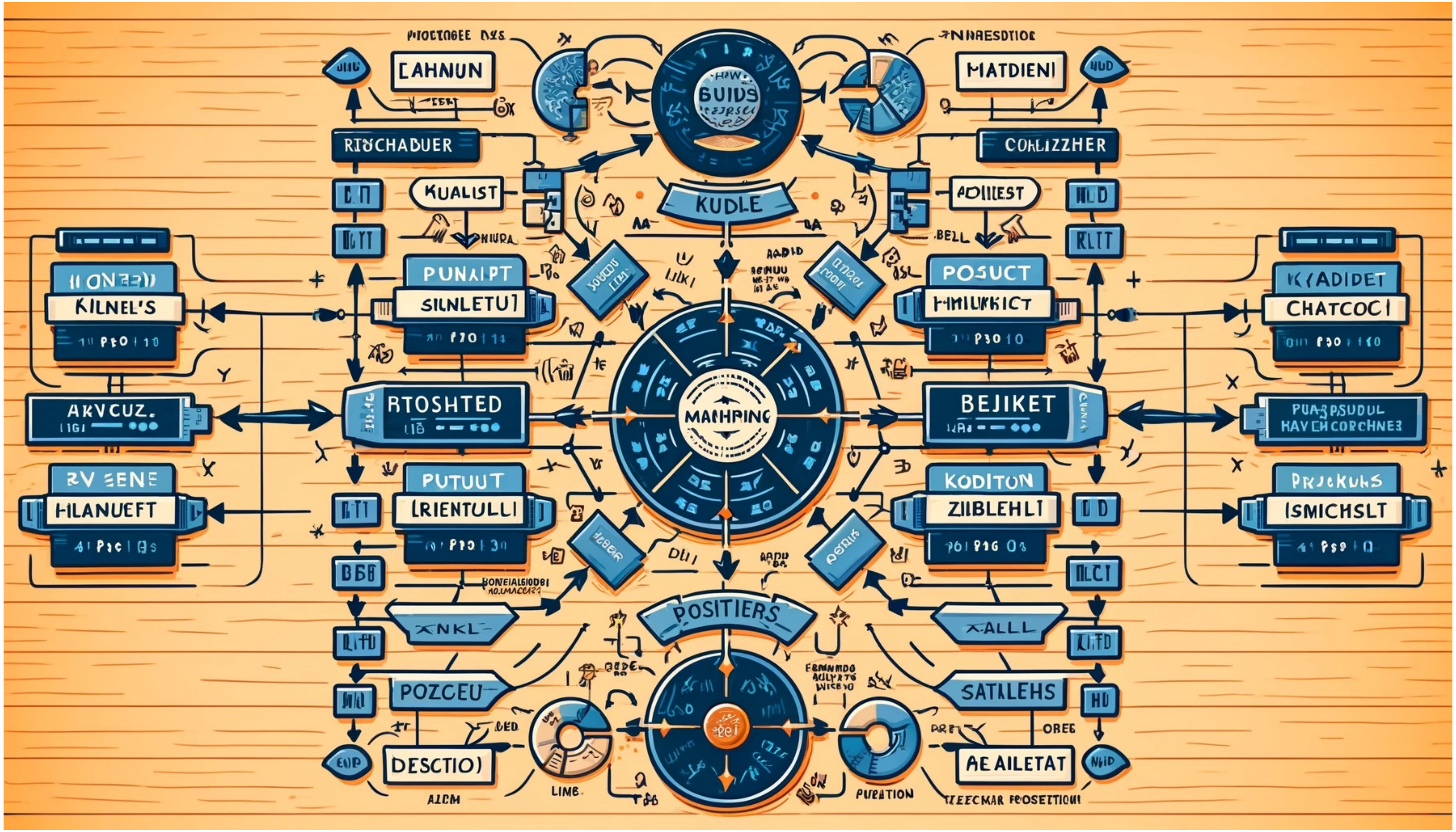
sql

Copy code

```sql
-- Load data into Hub_Kunde from Kunde
INSERT INTO Hub_Kunde (KundeID, Load_Date, Record_Source)
SELECT KundeID, CURRENT_TIMESTAMP, 'source_system_name' -- Replace with actual source
FROM Kunde;

-- Load data into Sat_Kunde from Kunde
INSERT INTO Sat_Kunde (KundeID, Load_Date, Record_Source, Vorname, Name, Geschlecht,
SELECT KundeID, CURRENT_TIMESTAMP, 'source_system_name', Vorname, Name, Geschlecht, G
FROM Kunde;
```

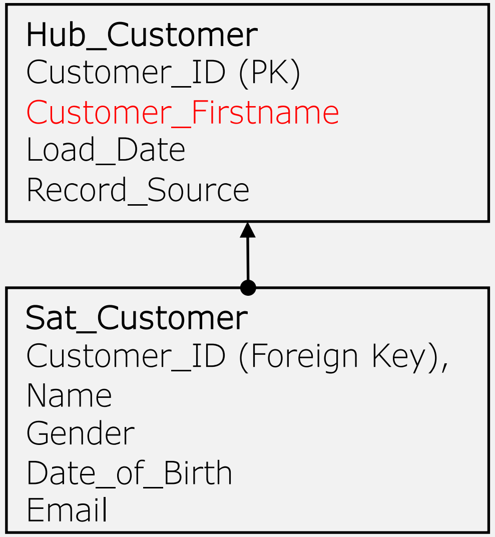
Original UML Field	Data Vault Entity	Data Vault Attribute
KundeID	Hub_Kunde	KundeID
Name	Sat_Kunde_Details	
Vorname	Sat_Kunde_Details	
Geschlecht	Sat_Kunde_Details	
Geburtsdatum	Sat_Kunde_Details	
...	...	
ProduktID	Hub_Produkt	
Bezeichnung	Sat_Produkt_Details	
Preis	Sat_Produkt_Details	


CAHNUN
BUIDS
MATDIENI
RIYCHADUER
COHIZZHER
KUALIST
KUDLE
ADIIEST
PUNAIPT
SINLETUI
POSUCT
HMIUKKICT
KILNEL'S
CHATCOCI
RTOSHTED
MATHPING
BEJIKET
AKVCUZ.
PUTUUT
LRIENTULLI
KODITON
ZIIBLEHLT
HLANUEFT
ISMICHSLT
POSITIERS
XNKL
XALLL
POZCEU
SATILEHS
DESCTIOI
AE AILETAT
LINE.
PURATION

We are interested in **optimizing an existing data vault model** and wish to get recommendations and explanations

TASK → PROMPT → RESULT

We give the model an existing but flawed data model:



The Firstname doesn't belong into the hub, the satellite is missing meta data fields

We prompted …

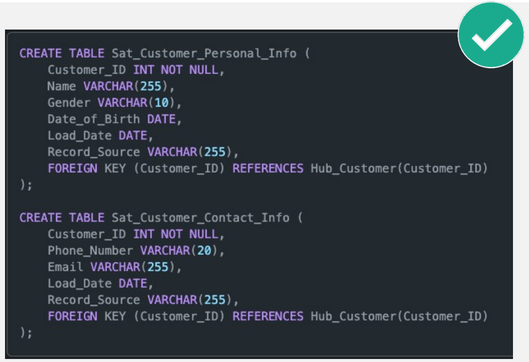
"What can be improved in this data vault model?"

"Create me the SQL statements to apply these changes"

"Why did you split the customer data into personal and contact information?"

✓ "2. Remove Firstname from Hub_Customer"
"3. Add Record Source to Satellite Tables"

✓



```
CREATE TABLE Sat_Customer_Personal_Info (
    Customer_ID INT NOT NULL,
    Name VARCHAR(255),
    Gender VARCHAR(10),
    Date_of_Birth DATE,
    Load_Date DATE,
    Record_Source VARCHAR(255),
    FOREIGN KEY (Customer_ID) REFERENCES Hub_Customer(Customer_ID)
);

CREATE TABLE Sat_Customer_Contact_Info (
    Customer_ID INT NOT NULL,
    Phone_Number VARCHAR(20),
    Email VARCHAR(255),
    Load_Date DATE,
    Record_Source VARCHAR(255),
    FOREIGN KEY (Customer_ID) REFERENCES Hub_Customer(Customer_ID)
);
```

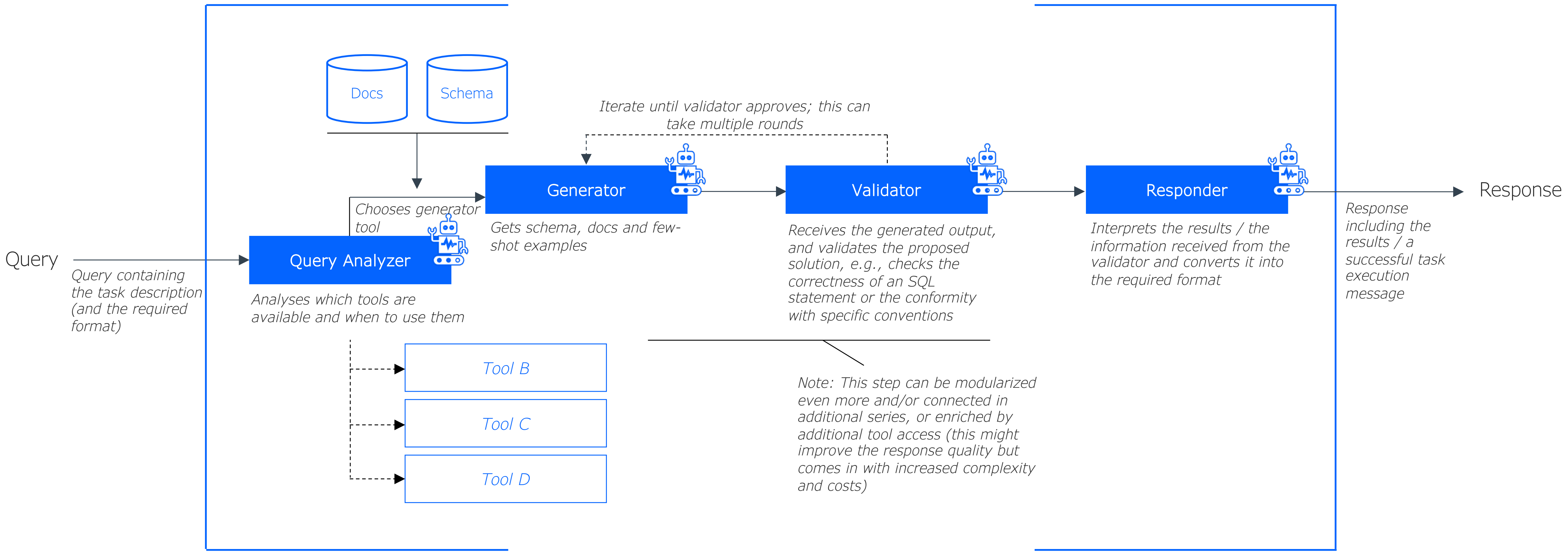
✓ "Improved Organization, Enhanced Flexibility, Better Performance, Increased Security, Easier Maintenance"

The GenAI automation toolbox offers a **variety of methods** to further **improve the quality and reliability** of the results

Method	Description	UseCases
Prompt Engineering	Add **specific goals**, commands or output formats to the prompt	▪ Narrow down the options, e.g. what exact model technique ▪ Prevent / guardrail certain unwanted outputs
Retrieval Augmented Generation	**Additional data** is retrieved from a (vector) database and handed in along with the query	▪ Internal data (e.g. from your knowledge base) shall be used ▪ Bigger amounts of text shall be considered
Large Action Models	Implements a **feedback cycle** provided by another model	▪ From simple compilation checks ▪ To complex validation via second LLM
Fine-Tuning the Foundation model	Use additional data to **train the foundation model** new knowledge	▪ Learn on domain knowledge ▪ Train on specific best-practice patterns

Large Action Models (LAMs) focus on **understanding actions and orchestrating sequences of actions** to accomplish specific goals without requiring assistance

Engineering Flow of a multi-stage, graph-based AI approach



Query

Query containing the task description (and the required format)

Docs

Schema

Query Analyzer

Analyses which tools are available and when to use them

Chooses generator tool

Tool B

Tool C

Tool D

Generator

Gets schema, docs and few-shot examples

Iterate until validator approves; this can take multiple rounds

Validator

Receives the generated output, and validates the proposed solution, e.g., checks the correctness of an SQL statement or the conformity with specific conventions

Note: This step can be modularized even more and/or connected in additional series, or enriched by additional tool access (this might improve the response quality but comes in with increased complexity and costs)

Responder

Interprets the results / the information received from the validator and converts it into the required format

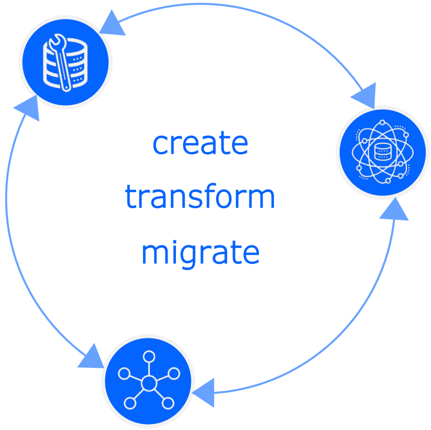
Response

Response including the results / a successful task execution message

Looking Ahead

What we expect from the development in the field of GenAI in 2024 and beyond

Our sample evaluation of general-purpose LLMs shows **impressive out-of-the-box functionality** on data modeling and sheds light on how **next level data modeling automation** might look like



3rd Normal Form (3NF)
according to Inmon

Write optimized

Star schema
according to Kimball

Read optimized

create

transform

migrate

Data Vault
according to Linstedt

Agile approach

We believe that **modeling, optimizing and migration** between data models is soom going to be **largely supported** by specialized tools

We have found **strengths and weaknesses with LLM’s** in the context of data management

Recommendations

Be **specific about your target** model to get meaningful results

Use tools that **close the gap** between your working environment and the LLM API

Anticipate the **cost aspect** before setting up a scalable solution

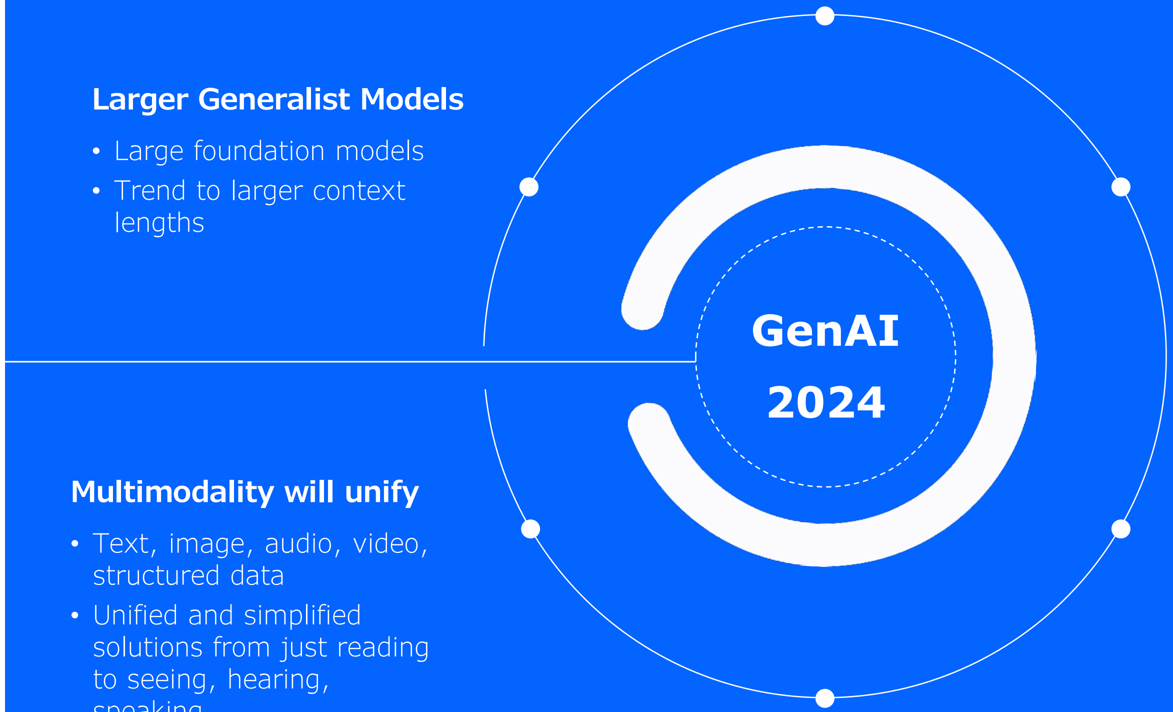
Leverage **LLMs to get you started** but do not rely on end-to-end solutions yet

Limitations

Don’t expect LLMs running in **business critical** production environments soon

Large-scale automation is not yet supported by tools – you need to built it yourself

We have **6 general hypotheses on GenAI** for the year 2024 and believe we are going to see **productionized AI features in DWA tools by end of this year**



Smaller, built-in Expert Models

- Model sizes decrease, while maintaining same performance
- Efficient, specialized models e.g. for data vault modeling in specific domains

Larger Generalist Models

- Large foundation models
- Trend to larger context lengths

Maintained Market Dominance of Big Tech

- OpenAI, Google, Meta, Anthropic, Mistral, etc.

Multimodality will unify

- Text, image, audio, video, structured data
- Unified and simplified solutions from just reading to seeing, hearing, speaking

New Skill Requirements

- Interaction with AI for all
- Not prompt engineer, but standard skill for everyone

Guardrailing & Security

- Protection against malicious attacks (e.g., prompt injection)
- Prevention from extraction of raw training data or other confidential information

eraneos