



# Coding Academy: Ein Smart Data Fabric für RAG

InterSystems Developer Roadshow



**Felix Vetter**  
Sales Engineer



**Stephan Mohr**  
Sales Engineer

# KI demystifiziert



## Artificial Intelligence

### Machine Learning

Image Recognition

Survival Analysis

...

### Tabular ML

Classification

Regression

Clustering

...

### Deep Learning

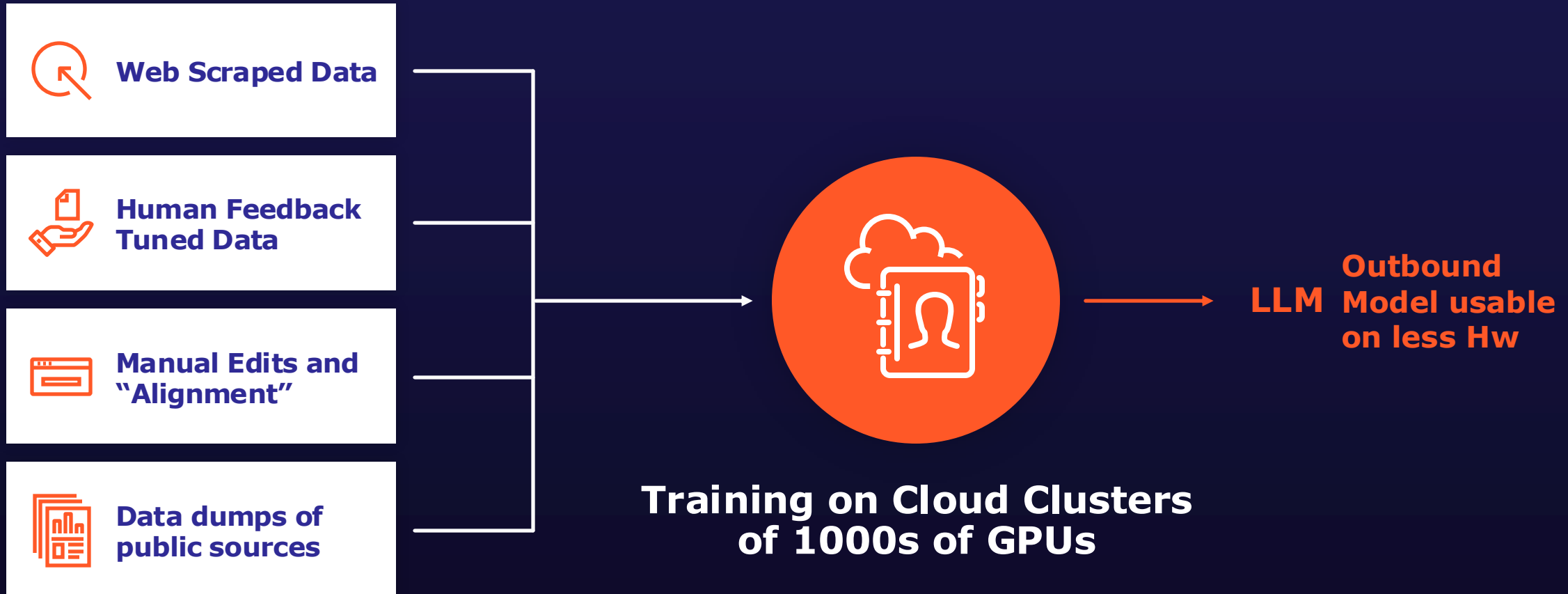
#### Generative AI

Large Language Models

Vector Search

...

# LLMs werden in einer VIELFALT an Quellen trainiert



# Kontext: Der Stand der LLM Apps



LLM haben fehlendes Fachwissen



Fine Tuning

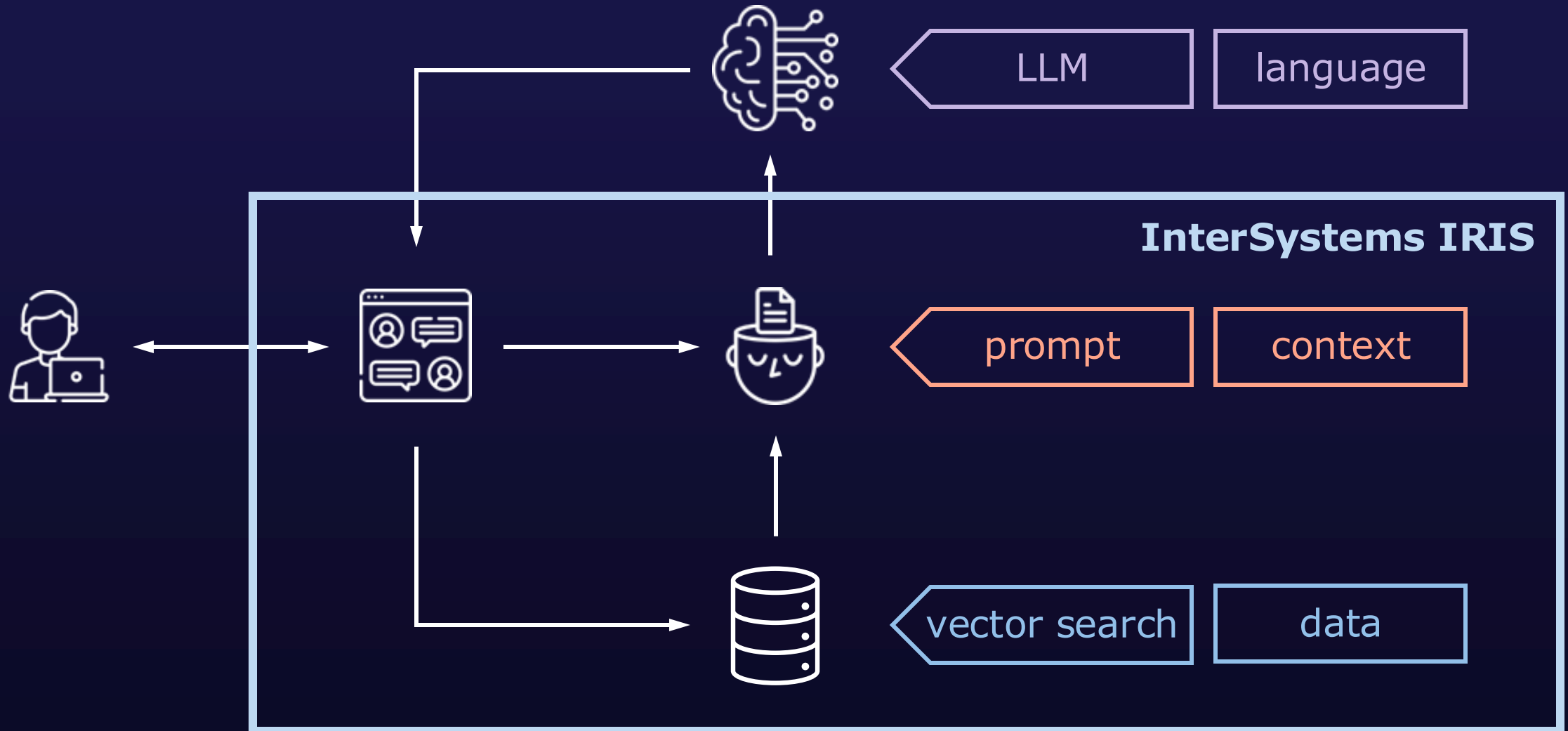


LLM hat keinen Zugriff zu privaten oder spezifischen Informationen

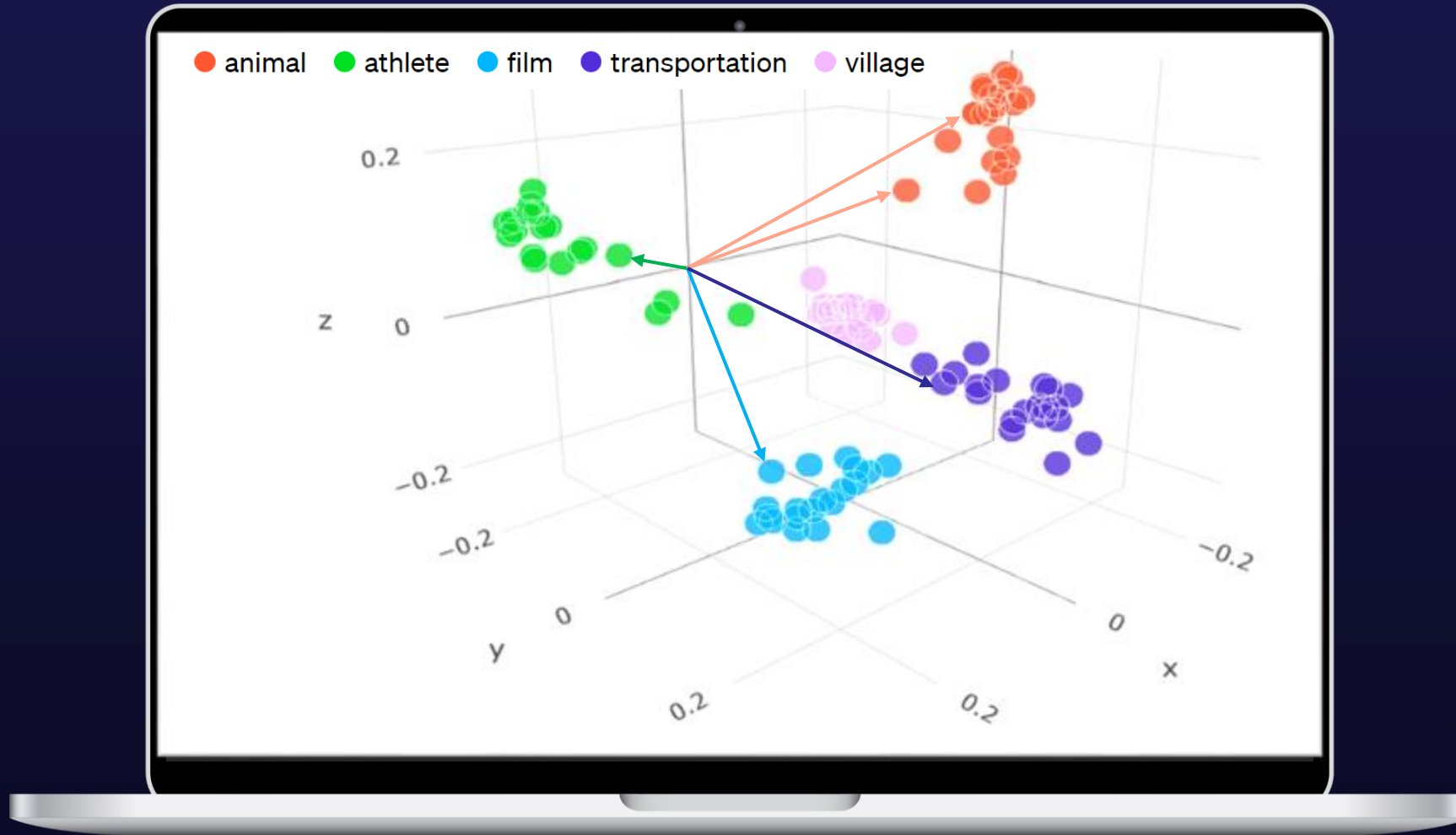


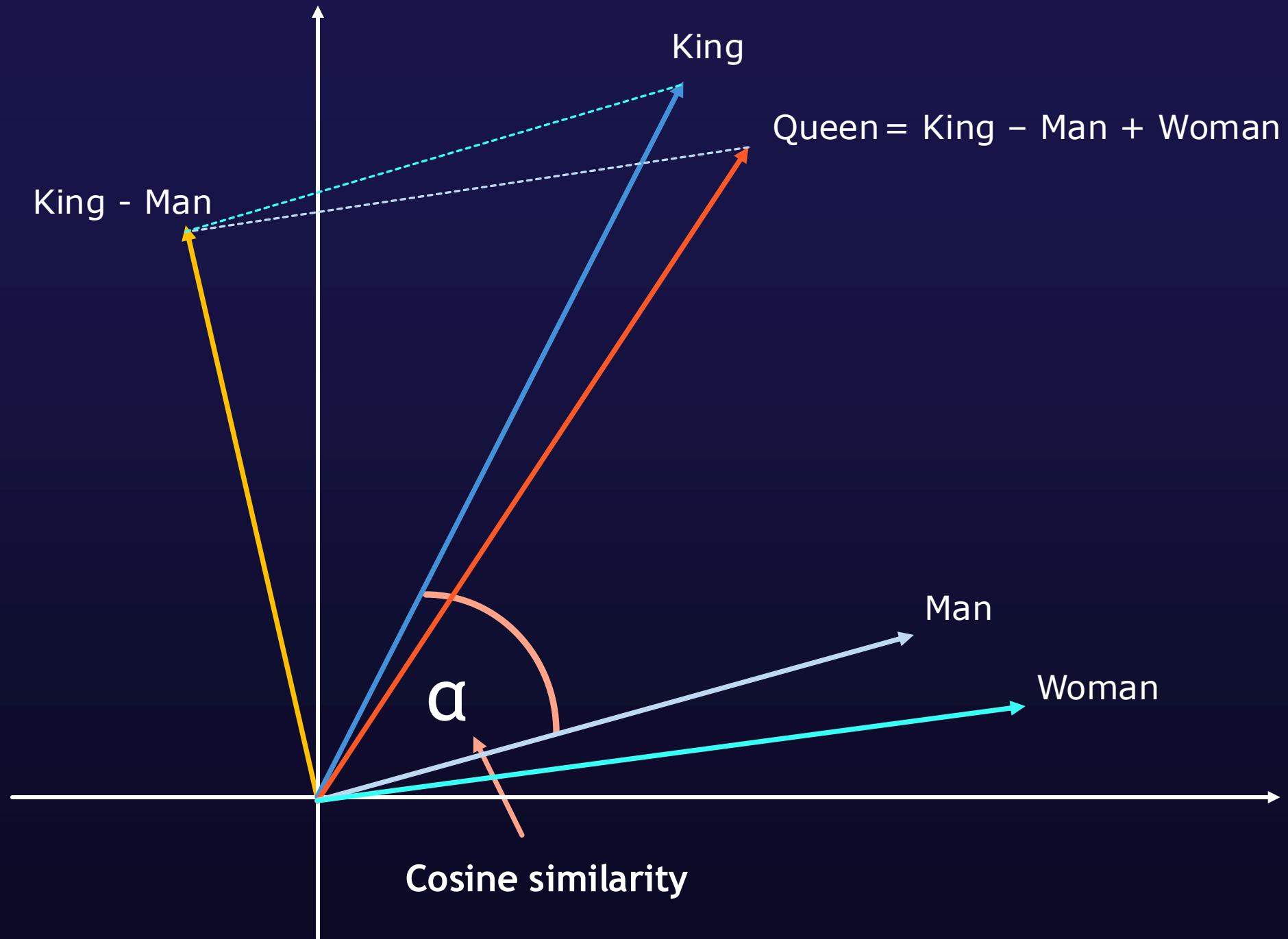
Retrieval Augmented Generation (RAG)

# RAG mit InterSystems IRIS

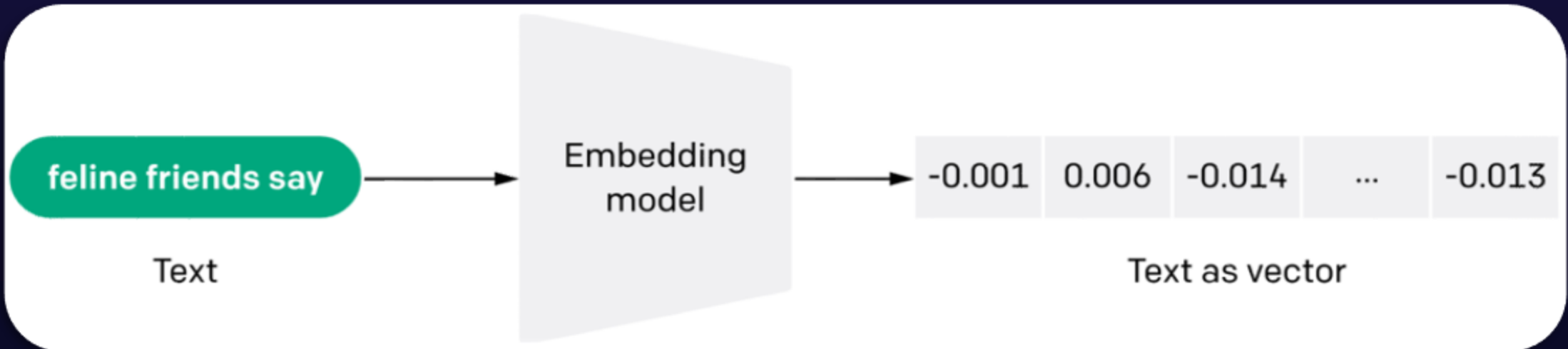
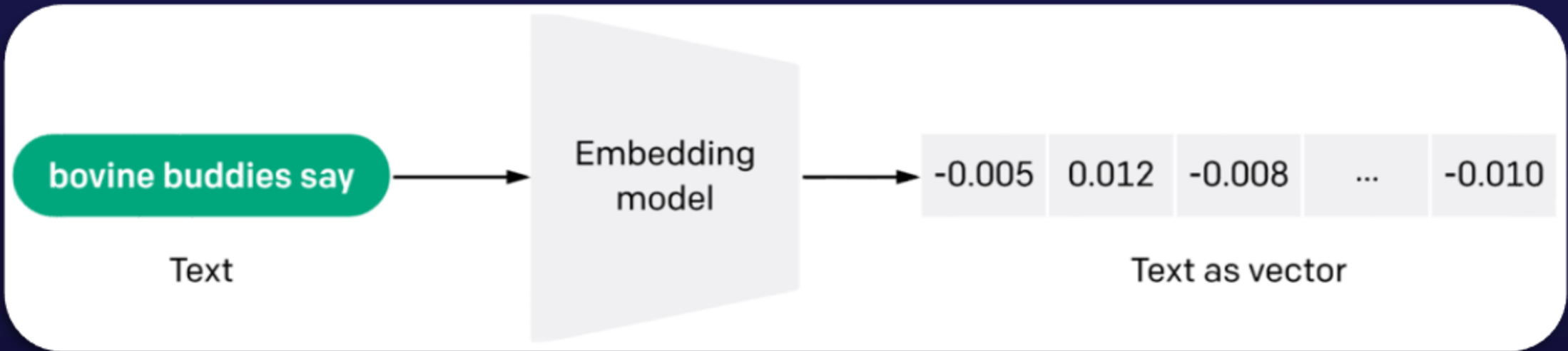


# Sätze & Kontext als Vektoren





# Word Embeddings





# Chunking und Overlapping



Splitter: Character Splitter  

Chunk Size: 31 

Chunk Overlap: 4 

Total Characters: 454

Number of chunks: 15

Average chunk size: 30.3

Die Logistikflotte von Logystic: Die Firma hat mehrere Flugzeuge. Lufttransporte Frachtflugzeuge:  
Eine Flotte von 20 Boeing 767-Frachtern und 10 Airbus A330-200F, die internationale Routen  
bedienen. Containerisierte mobile Lagerhäuser: Innovative mobile Lagereinheiten, die schnell in  
Katastrophengebieten eingesetzt oder während Spitzenzeiten zur Erhöhung der Lagerkapazität  
genutzt werden können.

Splitter: Character Splitter  

Chunk Size: 264 

Chunk Overlap: 66 

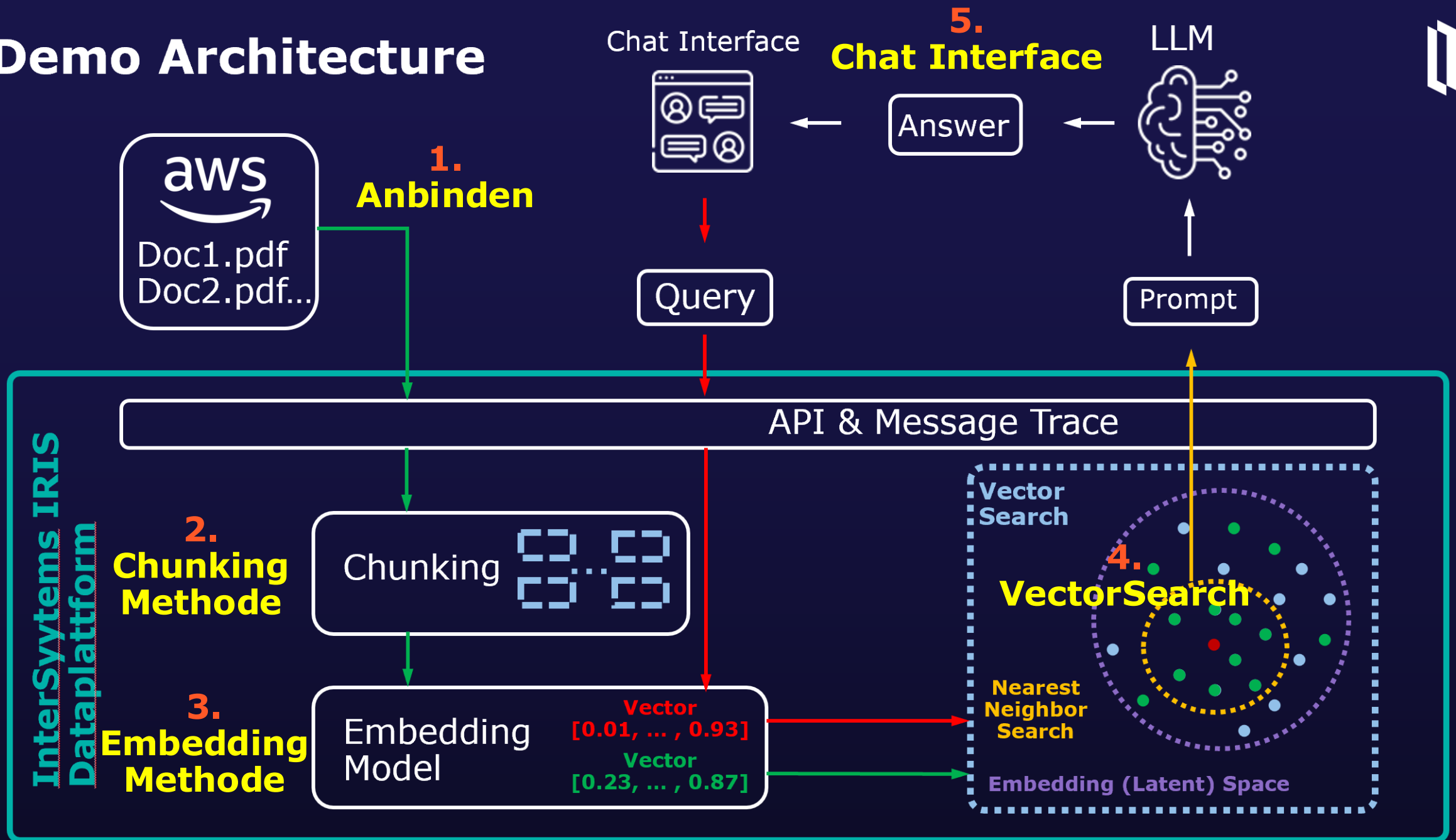
Total Characters: 464

Number of chunks: 2

Average chunk size: 232.0

Die Logistikflotte von Logystic: Die Firma hat mehrere Flugzeuge. Lufttransporte Frachtflugzeuge:  
Eine Flotte von 20 Boeing 767-Frachtern und 10 Airbus A330-200F, die internationale Routen  
bedienen. Containerisierte mobile Lagerhäuser: Innovative mobile Lagereinheiten, die schnell in  
Katastrophengebieten eingesetzt oder während Spitzenzeiten zur Erhöhung der Lagerkapazität  
genutzt werden können.

# Demo Architecture



# InterSystems IRIS Vector Search



- neuer VECTOR Datentyp & Ähnlichkeitssuche

```
CREATE TABLE t (txt VARCHAR(1000), vec VECTOR(INT, 200));
```

```
INSERT INTO t VALUES ('...', TO_VECTOR('1,2,3,...', INT));
```

```
SELECT TOP 10 * FROM  
FROM ( SELECT t.*, VECTOR_DOT_PRODUCT(vec, TO_VECTOR(...)) AS similarity FROM t )  
ORDER BY similarity DESC;
```

- Ähnlichkeitssuche ist die Grundlage von RAG
  - Dennoch nützlich auch außerhalb von GenAI

# Zeitplan



## Vector Search Early Access Program

- Eingeführt, als Teil der 2024.1 Developer Preview
- Enthält VECTOR Datentyp und SQL Funktionen



## 2024.1 GA – März 2024

- VectorSearch, SIMD-gestützt für die schnelle Suche in Vektoren

## 2024.3 GA – Ende von 2024

- Embeddings Index, ANN-gestützt für die schnelle Suche in Vektoren



# Aufgabe 0: Verbindung von S3 zu IRIS



```
if(self.Bucket != "" and self.Key != ""):

    from io import BytesIO
    from PyPDF2 import PdfReader
    import boto3
    from botocore import UNSIGNED
    from botocore.client import Config
    from botocore.exceptions import ClientError

    # Initialize a session using Amazon S3 with unsigned configuration
    s3 = boto3.client('s3', config=Config(signature_version=UNSIGNED))

    response = s3.get_object(Bucket=self.Bucket,Key=self.Key)

    reader = PdfReader(BytesIO(response['Body'].read()))

    text = ""

    for page in reader.pages:
        text += page.extract_text()

    return text
else:
    return ""
```



# Aufgabe 1: Die GetChunks() Methode



*## Utils.cls Method GetChunks*

```
from langchain.text_splitter import RecursiveCharacterTextSplitter
```

```
text_splitter = RecursiveCharacterTextSplitter(  
    chunk_size = pChunkSize,  
    chunk_overlap = pChunkOverlap  
)
```

```
docs = text_splitter.split_text(pText)
```

```
return docs
```



## Aufgabe 2: Die GetEmbeddingPy() Methode

- Gehe zu [Docs.Intersystems.com](https://docs.intersystems.com) und suche nach GetEmbeddings() Methode
- Kopiere es in die Demo.Utills Klasse

```
## Utils.cls Method GetEmbeddingPy

import json

# import the package
import sentence_transformers

# create the model and form the embeddings
model = sentence_transformers.SentenceTransformer('all-MiniLM-L6-v2')
embeddings = model.encode(sentences)

# convert the embeddings to a string
embeddings_list = [str(embedding.tolist()) for embedding in embeddings]
# print(embeddings_list[0])
return embeddings_list
```



## Aufgabe 3: Definiere das Property

- Gehe zu Demo.RecordEmbeddings.cls
- Erstelle ein Property names "Embedding" als Type %Vector.
- Die Länge sollte 384 sein.



```
Property Embedding As %Vector(LEN = 384);
```





## Aufgabe 4: Definiere das SQL Statement

Baue das SQL Statement:

- Insert into Demo.RecordEmbeddings
- Wir wollen in DataSourceId,SourceId,Text,Embedding abspeichern
- Die Werte sind (tId, tRecordId, tChunk, tVector) **Beachte Speicherung von tVector**



```
&sql(insert into Demo.RecordEmbeddings (DataSourceId,SourceId,Text,Embedding)  
      values (:tId,:tRecordId,:tChunk,TO_VECTOR(:tVector)))
```

# Aufgabe 5: Das Chat-Interface




Developer Roadshow

localhost:8501

Message Viewer SQL Production Configu... InterSystems IRIS D...

Deploy

# Welcome to Developer Roadshow



**InterSystems®**  
Creative data technology

User: How many boeings does logistics have? Only write the number

LLM: 20

You:



## Aufgabe 6: Die Vector Search implementieren

Baue das SQL Statement:

- Select the TOP 5 ID from the Demo.RecordEmbeddings
- Ordne absteigend nach dem Ergebnis von VectorSearch(Embedding, tVector)



```
SELECT TOP 5 ID FROM Demo.RecordEmbeddings  
ORDER BY VECTOR_DOT_PRODUCT(Embedding, TO_VECTOR(?)) DESC
```

Demo.Operations.Prompt.PromptOperation

## Aufgabe 7: Die Prompt Operation verstehen



Demo.Operations.OpenAI.APIOperation

## Aufgabe 8: Die OpenAI Operation verstehen

## Aufgabe 9: Starte die Production

# Aufgabe 10: Starten des StreamlitUI Servers



- Öffne den Developer Roadshow Ordner auf dem Desktop
- Rechts Klick "Im Terminal öffnen"
- Führe "python -m streamlit run app.py" aus

Das UI sollte sich im Browser öffnen.

**Frage an den Chatbot: "How many Airplanes does Logystics have?"**

# Aufgabe 11: Wir brauchen Kontext



- Starte den "Start PDF Import" Service durch Doppelklick auf den Service

**Production Configuration** Start Stop

Production Stopped Category: All Legend Production Settings

Services +	Processes +	Operations +
<input type="radio"/> Start PDF Import	<input type="radio"/> Injection Process	<input type="radio"/> Embedding Operation
<input type="radio"/> Streamlit Service	<input type="radio"/> LLM Router	<input type="radio"/> OpenAI Operation
		<input type="radio"/> PDF Operation
		<input type="radio"/> Prompt Operation
		<input type="radio"/> RAG Operation

# Aufgabe 12: Let's RAG



- Frage an den Chatbot erneut:

**“How many Airplanes does Logystics have?”**

- Frage an den Chatbot:

**“Who is the venture capitalist and how much was invested?”**

# Thank You & Let's Connect!



**Felix Vetter**

Sales Engineer

Felix.Vetter@intersystems.com



**Stephan Mohr**

Sales Engineer

Stephan.Mohr@intersystems.com