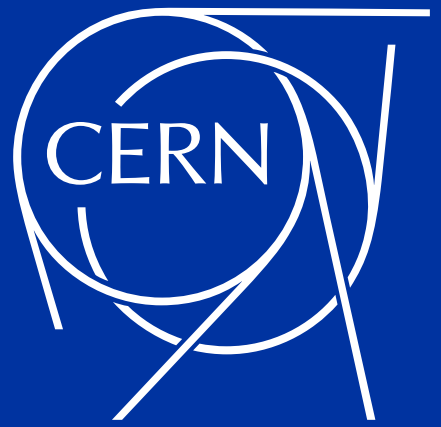


Select AI
How to query your data using natural language

Andrzej Nowicki

SOUG, Zurich, 2025

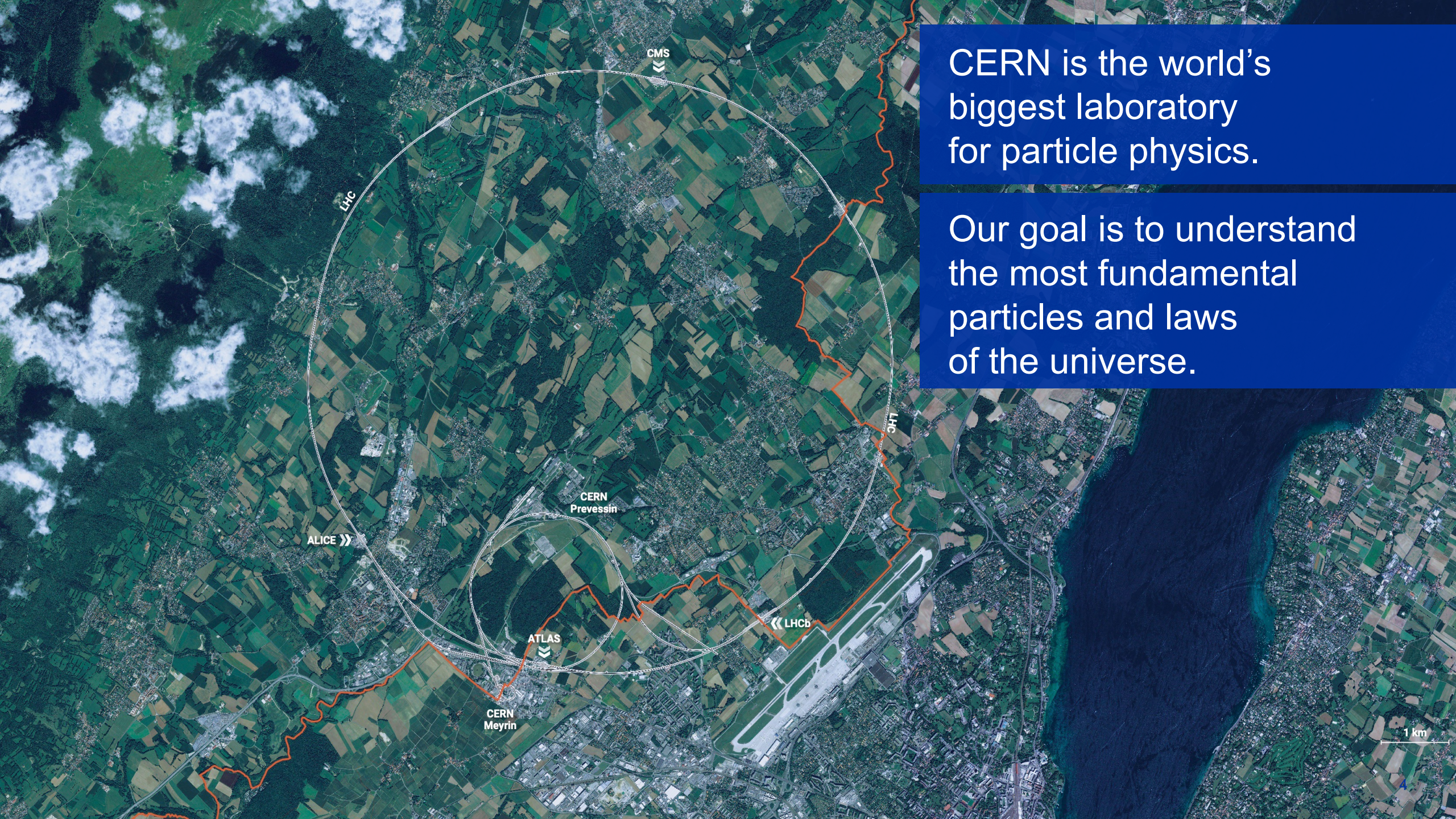
Andrzej Nowicki

13 years of Oracle DB experience
Database Engineer @ CERN since 2020

andrzejnowicki

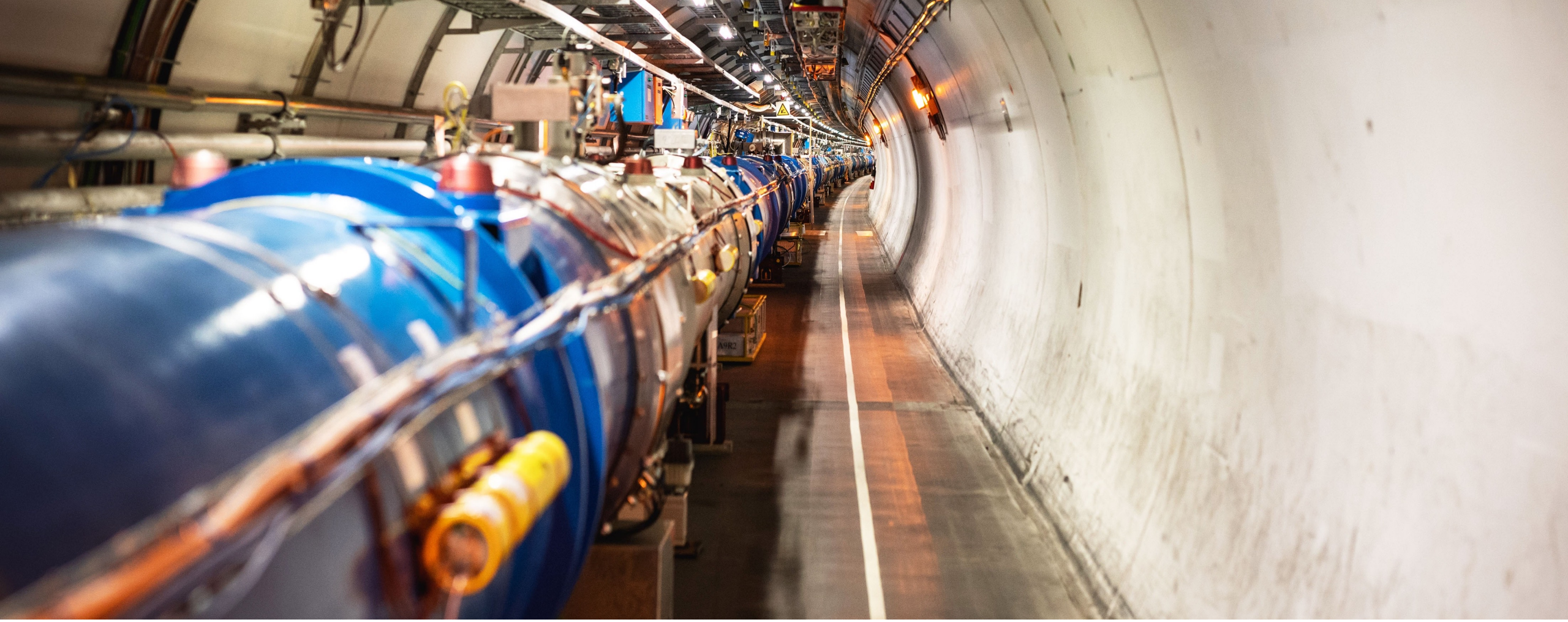
andrzej.nowicki@cern.ch

www.andrzejnowicki.pl

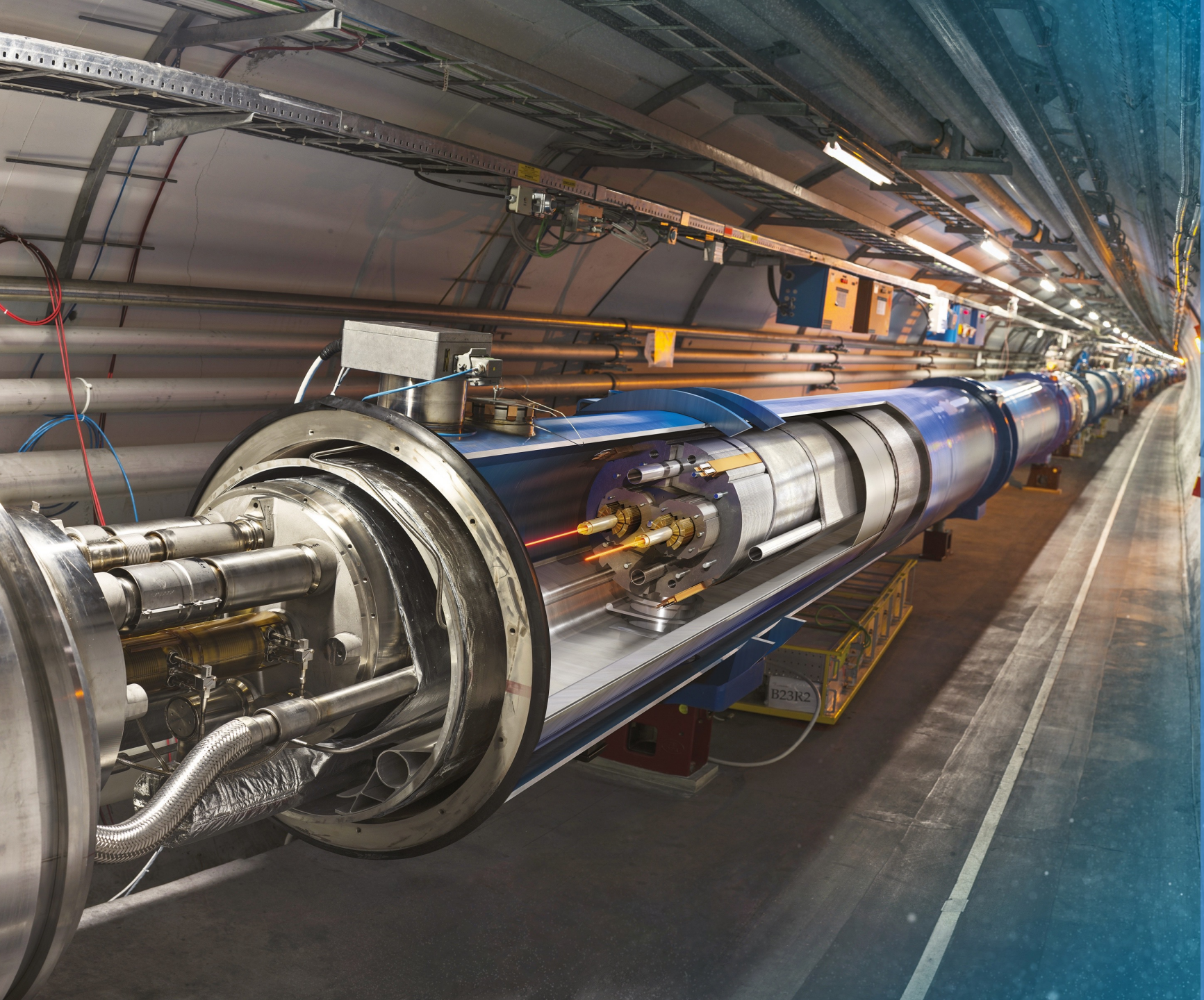


CERN is the world's biggest laboratory for particle physics.

Our goal is to understand the most fundamental particles and laws of the universe.

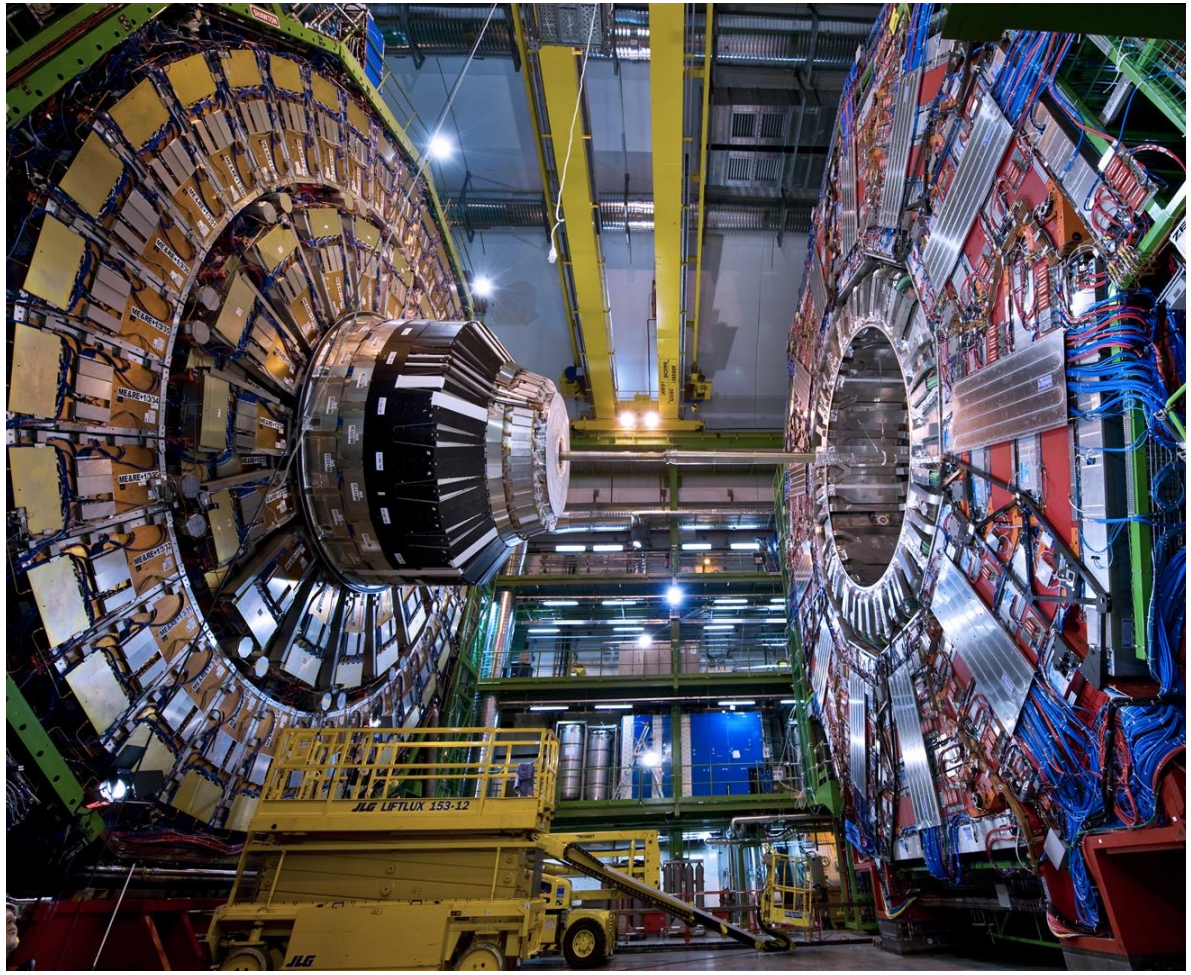


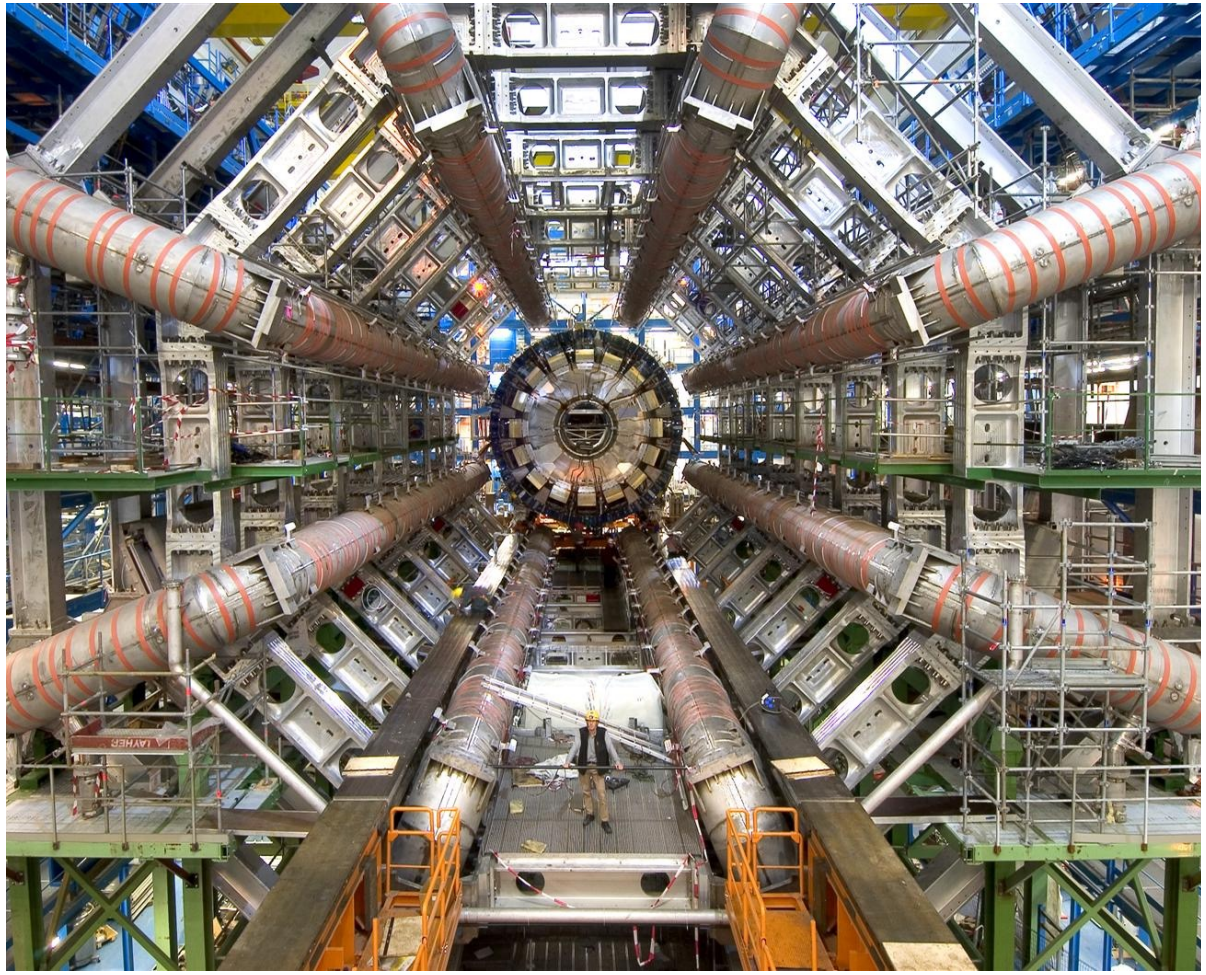
Large Hadron Collider (LHC)



Large Hadron Collider (LHC)

- 27 km (17 mi) in circumference
- About 100 m (300 ft) underground
- Superconducting magnets steer the particles around the ring
- Particles are accelerated to close to the speed of light


JLG LIFTLUX 153-12
JLG



IT @ CERN

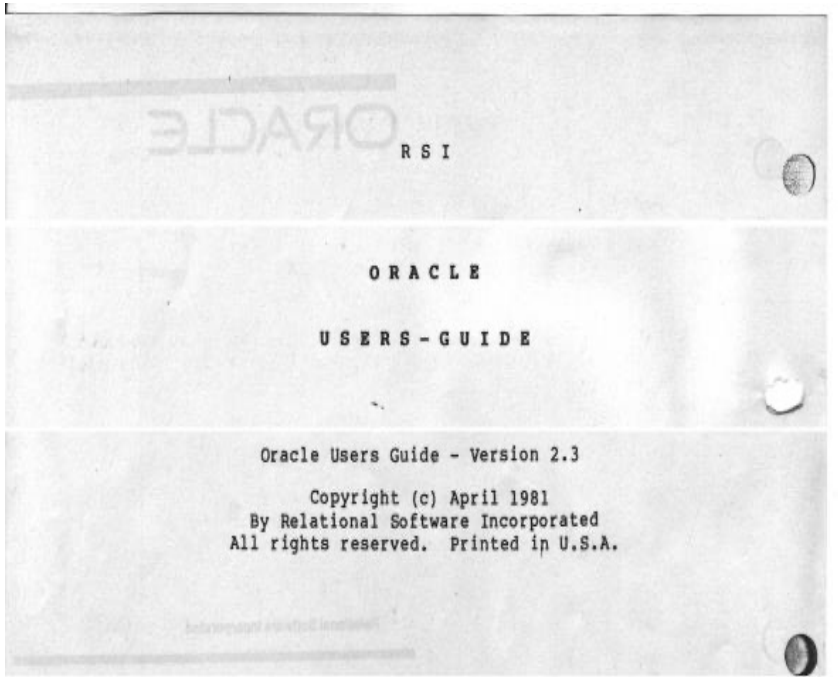
Databases at CERN

Oracle since 1982

- 105 Oracle databases, more than 11.800 Oracle accounts
- RAC, Active Data Guard, GoldenGate, OEM, RMAN, APEX, Cloud…
- Complex environment

Database on Demand (DBoD) since 2011

- ≈600 MySQL, ≈400 PostgreSQL, ≈200 InfluxDB
- Automated backup and recovery services, monitoring, clones, replicas
- HA MySQL clusters (Proxy + primary replica)



Oracle 23ai

WOW!

Oracle 26ai

Where to test 26ai?

I'll be using the **Always Free** Autonomous Database **…**

Autonomous Database with Oracle Database 23ai in **the Paid tier** is available in all commercial public cloud regions.

Always Free Autonomous Database with Oracle Database 23ai is available in all commercial public cloud regions ***except*** the following regions: Colombia Central: Bogota (BOG), Saudi Arabia Central (RUH), Singapore West: Singapore (XSP)

Beware! The calls to GenAI service are not free

https://docs.oracle.com/en-us/iaas/autonomous-database-serverless/doc/autonomous-always-free-23ai.html

Feel free to take photos, but…

The presentation is on my website

https://www.andrzejnowicki.pl/slides/

SELECT AI
is only available in the cloud

* available on Exadata platforms as of 23.7

Select AI

Querying data using natural language, conversations

- Generating SQL statements
- Explaining SQL statements

Synthetic data generation

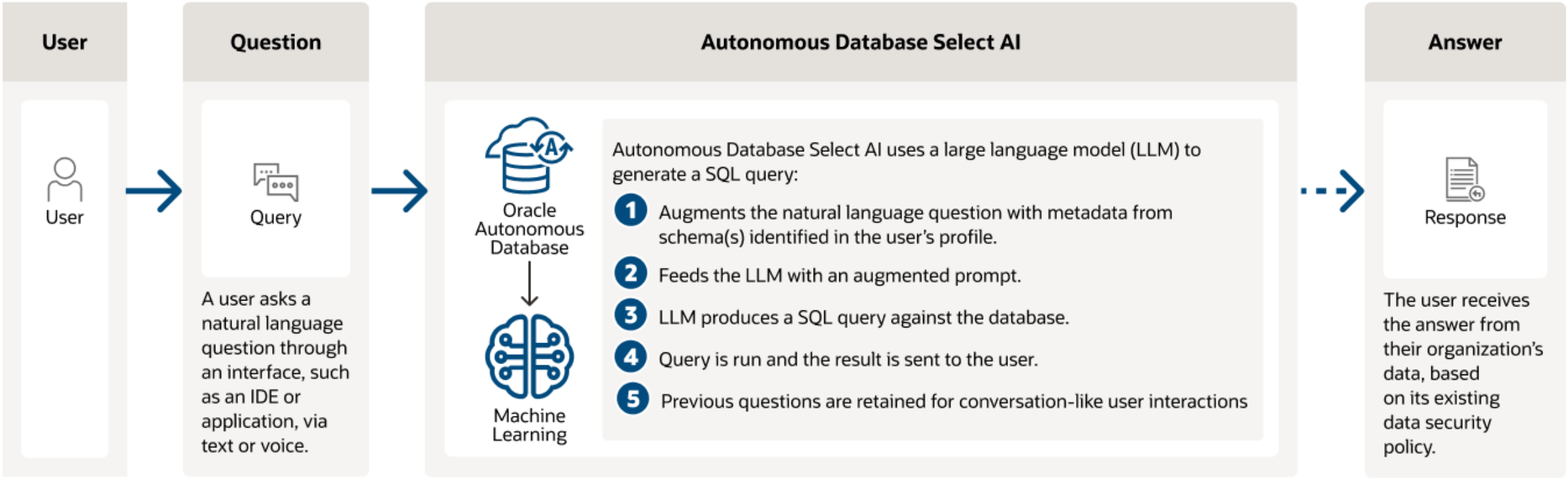
RAG systems – Retrieval Augmented Generation

As of September/October, there's the addition of Select AI Translate and Agents.
I'm unfortunately not going to cover that

Select AI – querying data

```
SQL> SELECT AI What are total sales of tom hanks movies
70,318.23

SQL> SELECT AI showsql What are total sales of tom hanks movies
SELECT SUM(sales_sample.list_price) AS total_sales
FROM moviestream.sales_sample
JOIN moviestream.movie ON sales_sample.movie_id = movie.movie_id
WHERE movie.cast LIKE '%Tom Hanks%'
```

Select AI – querying data



https://www.oracle.com/autonomous-database/select-ai/

AI features – credentials

```
SQL> BEGIN
     DBMS_CLOUD.CREATE_CREDENTIAL (
       credential_name => 'OCI_CRED',
       user_ocid       => 'ocid1.user.oc1...',
       tenancy_ocid    => 'ocid1.tenancy.oc1.....',
       private_key     => 'MII......=',
       fingerprint     => '44:.......');
END;
/
```

AI features – profile

```
SQL> BEGIN
     DBMS_CLOUD_AI.create_profile(
      'OCI',
      '{"provider": "oci",
        "credential_name": "OCI_CRED",
        "oci_compartment_id": "ocid1.compartment.oc1......",
        "region" :"eu-frankfurt-1",
        "object_list": [{"owner": "ADMIN", "name": "employees"}]
       }');
END;
/
```

The object's data is not shared to the LLM.
Only metadata (column definitions, etc.)

```
SQL> select  * from employees;

        ID NAME           SALARY
---------- ---------- ----------
        21 Christi           100
        22 Andrzej           101
        23 Anna              999
```

AI features – examples

```
SQL> EXEC DBMS_CLOUD_AI.set_profile('OCI');

SQL> select ai chat how many people live in Poland;

RESPONSE
--------------------------------------------------------------------------------
As of 2021, the estimated population of Poland is approximately 38.6 million people

SQL> select ai how many employees do we have;

EmployeeCount
-------------
            3
```

AI features – examples

```
SQL> select ai showsql how many employees do we have;

RESPONSE
--------------------------------------------------------------------------------
SELECT COUNT(c."ID") AS "EmployeeCount"
FROM "ADMIN"."EMPLOYEES" e


SQL> select ai narrate how many employees do we have;

RESPONSE
--------------------
We have 3 employees.
```

🚨 This exposes your data to the LLM !

DEMO

AI features – limitation – we don't see the SQL

```
SQL> select ai how many employees do we have;
SQL> select ai showsql how many employees do we have;
```

These are 2 separate calls to LLM
You can get different results

```
SQL> SELECT sql_fulltext
     FROM v$sql
     WHERE sql_id = (
       SELECT prev_sql_id
       FROM   v$session
       WHERE  audsid = SYS_CONTEXT('USERENV', 'SESSIONID')
);
```

Autotrace does not work

```
SQL> WITH sqlid AS (
  SELECT prev_sql_id
  FROM   v$session
  WHERE  audsid = SYS_CONTEXT('USERENV', 'SESSIONID')
)
SELECT s.sql_fulltext
FROM v$sql s
JOIN sqlid i ON s.sql_id = i.prev_sql_id;
```

AI features – flexibility

```
SQL> BEGIN
  DBMS_CLOUD_AI.create_profile(
   'OCI_HR_COHERE',
   '{"provider": "oci",
     "model": "cohere.command-r-08-2024",
     "credential_name": "OCI_CRED_",
     "oci_compartment_id": "oci...",
     "region" :"eu-frankfurt-1",
     "conversation": "true",
     "object_list": [{"owner": "HR"}]
    }');
END;
/
```

```
SQL> BEGIN
  DBMS_CLOUD_AI.create_profile(
   'OCI_HR_LLAMA',
   '{"provider": "oci",
     "model": "meta.llama-3.1-70b-instruct",
     "credential_name": "OCI_CRED_",
     "oci_compartment_id": "oci...",
     "region" :"eu-frankfurt-1",
     "conversation": "true",
     "object_list": [{"owner": "HR"}]
    }');
END;
/
```

Multiple providers supported:
OpenAI, Cohere, Azure OpenAI, OCI GenAI, Google, Anthropic, Hugging Face

Multiple models supported:
Llama, GPT, Cohere Command, Gemini, Claude

The default for OCI is meta.llama-3.3-70b-instruct

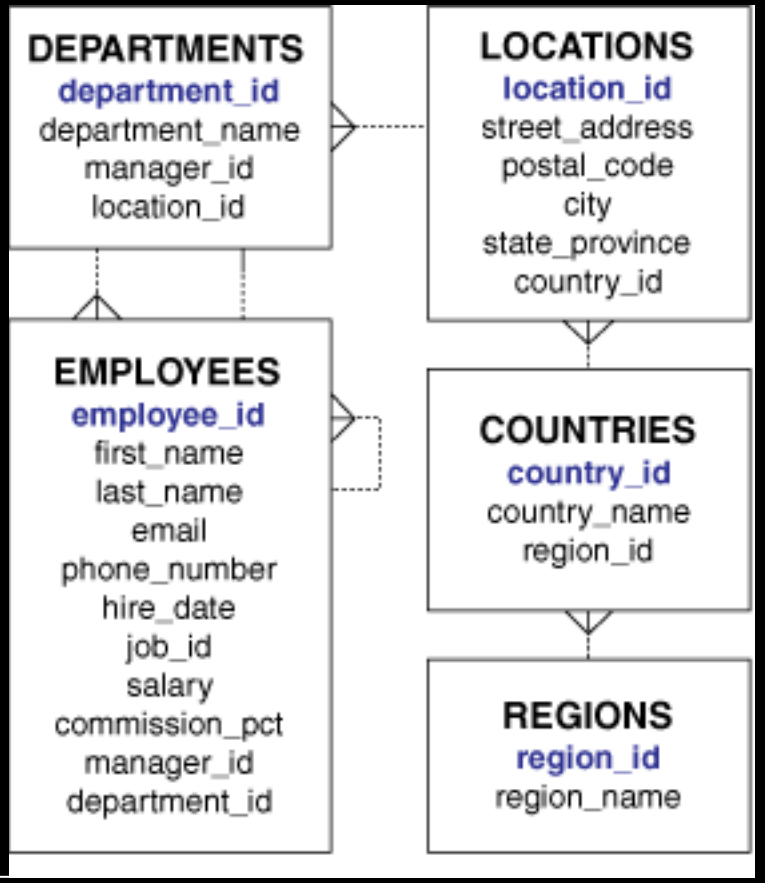
https://docs.oracle.com/en-us/iaas/autonomous-database-serverless/doc/select-ai-about.html#GUID-FDAEF22A-5DDF-4BAE-A465-C1D568C75812

AI features – flexibility

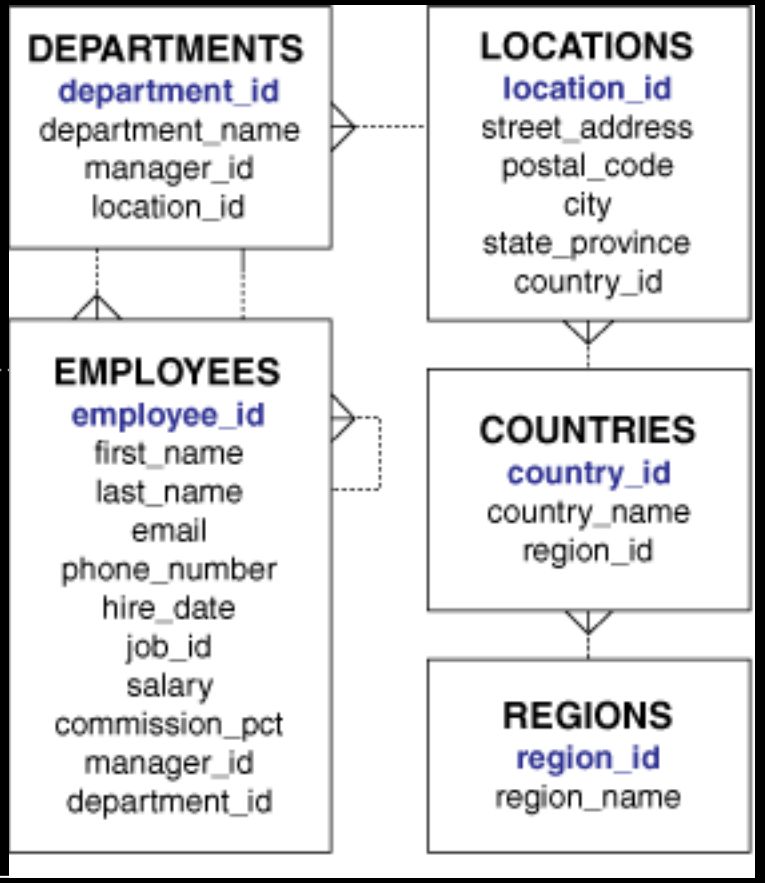
```
SQL> BEGIN
  DBMS_CLOUD_AI.create_profile(
   'OCI_HR_COHERE',
   '{"provider": "oci",
     "model": "cohere.command-r-08-2024",
     "credential_name": "OCI_CRED_",
     "oci_compartment_id": "oci...",
     "region" :"eu-frankfurt-1",
     "conversation": "true",
     "object_list": [{"owner": "HR"}],
     "max_tokens": 1024,
     "temperature": 0
    }');
END;
/
```

Parameters allow to modify the behaviour of the LLM model

In which regions do we have departments?



```
SQL> EXEC DBMS_CLOUD_AI.set_profile('OCI_HR_DEFAULT');

PL/SQL procedure successfully completed.

SQL> select ai showsql in which regions do we have departments?;

RESPONSE
--------------------------------------------------------------------------------
SELECT DISTINCT T2."REGION_NAME" FROM "HR"."DEPARTMENTS" T1 INNER JOIN "HR"."REGIONS" T2 ON T1."LOCATION_ID" = T2."REGION_ID"


SQL> EXEC DBMS_CLOUD_AI.set_profile('OCI_HR_COHERE');

PL/SQL procedure successfully completed.

SQL> select ai showsql in which regions do we have departments?;

RESPONSE
--------------------------------------------------------------------------------
SELECT r."REGION_NAME" AS "Region", COUNT(DISTINCT d."DEPARTMENT_ID") AS "Number of Departments"
FROM "HR"."REGIONS" r
LEFT JOIN "HR"."DEPARTMENTS" d ON r."REGION_ID" = d."MANAGER_ID"
GROUP BY r."REGION_NAME"
```



As of September, it's possible to add "feedback" on generated SQLs.
Both positive and negative.

AI features – flexibility

```
SQL> BEGIN
  DBMS_CLOUD_AI.create_profile(
   'OCI_HR',
   '{"provider": "oci",
     "credential_name": "OCI_CRED_",
     "oci_compartment_id": "...",
     "region" :"eu-frankfurt-1",
      "conversation": "true",
     "object_list": [{"owner": "HR"}]
    }');
END;
/
```

Up to 10 past prompts can be included in your prompt

```
> select ai in which regions do we have departments?;

no rows selected
```

Making typos helps sometimes

```
> select ai in which regions, countriedo we have departments?;

REGION_NAME              COUNTRY_NAME
------------------------ --------------------------------
Europe                   Germany
Americas                 Canada
...

> select ai in which regions, do we have departments?;

REGION_NAME
-------------------------
Europe
Americas
```

AI features – PL/SQL

```
SQL> select dbms_cloud_ai.generate('which AI model am I talking to? be precise',
                                   action=>'chat')
```

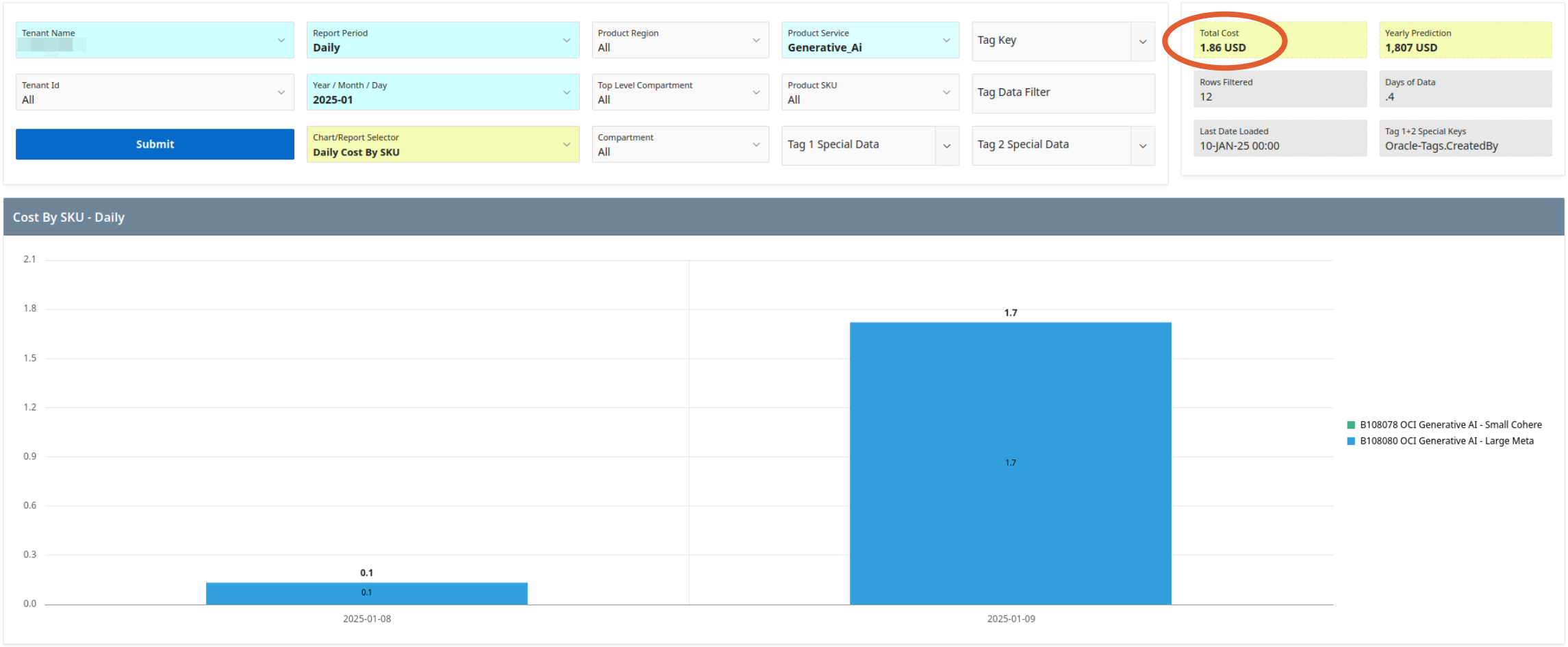
You are talking to a variant of the Meta AI model, specifically the LLAMA (Large Language Model Meta AI) model.
However, please note that I'm an instance of a model that is based on a specific snapshot of the LLAMA model, and my training data is current up to 2023.

AI features – PL/SQL

```
SQL> select dbms_cloud_ai.generate(
               'think of a nickname based on this bio, '||
               'return the nickname only:'||bio,
               action=> 'chat') bio
       from speakers
       where rownum <= 10;
```

DEMO

Select AI – costs?



Select AI

Querying data using natural language, conversations

- Generating SQL statements
- Explaining SQL statements

Synthetic data generation

RAG systems – Retrieval Augmented Generation

Synthetic data generation

Generate synthetic data using random generators, algorithms, statistical models, and Large Language Models (LLMs) to simulate real data for developing and testing solutions effectively.

🚨 This exposes your data to the LLM !

Synthetic data generation

```
create table brewery(id number primary key,
                     name varchar2(100),
                     country varchar2(30));

create table beer(id number primary key,
                  id_brewery number constraint beer_brewery_fk references brewery(id),
                  name varchar2(80),
                  description varchar2(200));

create table users(id number primary key,
                   name varchar2(10));

create table user_reviews(id number primary key,
                          id_user number constraint user_reviews_user_fk references users(id),
                          id_beer number constraint user_reviews_beer_fk references beer(id),
                          review varchar2(200));
```

Synthetic data generation

```
BEGIN
  DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA(
    profile_name => 'OCI',
    object_list =>
      '[{"owner": "ADMIN", "name": "BREWERY",
        "record_count":10},
       {"owner": "ADMIN", "name": "BEER",
        "user_prompt":"different styles of beer; creative and catchy name; marketing description",
        "record_count":50},
       {"owner": "ADMIN", "name": "USERS",
        "user_prompt":"funny usernames",
        "record_count":10},
       {"owner": "ADMIN", "name": "USER_REVIEWS",
        "user_prompt":"reviews from dissatisfied customers",
        "record_count":10}]'
  );
END;
/
```

Synthetic data generation

```
BEGIN
  DBMS_CLOUD_AI.GENERATE_SYNTHETIC_DATA(
    profile_name => 'OCI',
    object_name  => 'MOVIES',
    owner_name   => 'ADMIN',
    record_count => 10,
    params       => '{"sample_rows":5}'
  );
END;
/
```

DEMO

Select AI

Querying data using natural language, conversations

- Generating SQL statements
- Explaining SQL statements

Synthetic data generation

RAG systems – Retrieval Augmented Generation

A little bit of theory...

VECTORS

In AI, a vector is an ordered list of numbers (scalars) that can represent a point in a multidimensional space. Mathematically, a vector is often written as:

$$\mathbf{v} = (v_1, v_2, \ldots, v_{n-1}, v_n)$$

n is the dimensionality of the vector.

How to put it all together?

Input

(movie, picture, text, etc.)

Embedding model

That's a simplification.
Normally you would cut the text in chunks and embed each chunk separately

Embedding

Vector

"Citrusy, sweet aroma"

[0.329, 0.911, 0.21, 0.37, …]

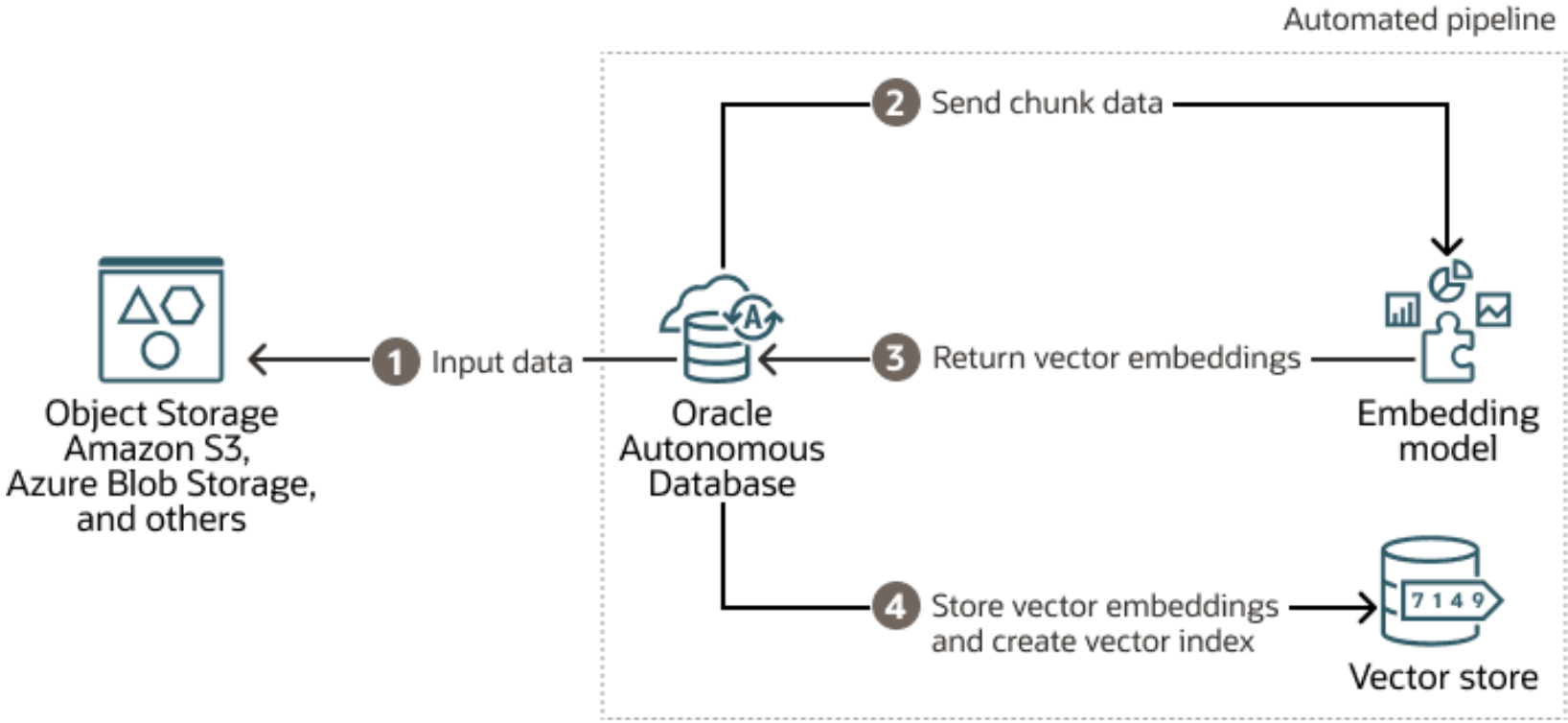
How to put it all together?

“Citrusy, sweet aroma”	`[0.329, 0.911, 0.21, 0.37, …]`
“Grapefruity taste, sweet aroma”	`[0.317, 0.818, 0.11, 0.36, …]`
“Harsh, spicy, roasted”	`[ 0.11,  0.01, 0.91, 0.87, …]`

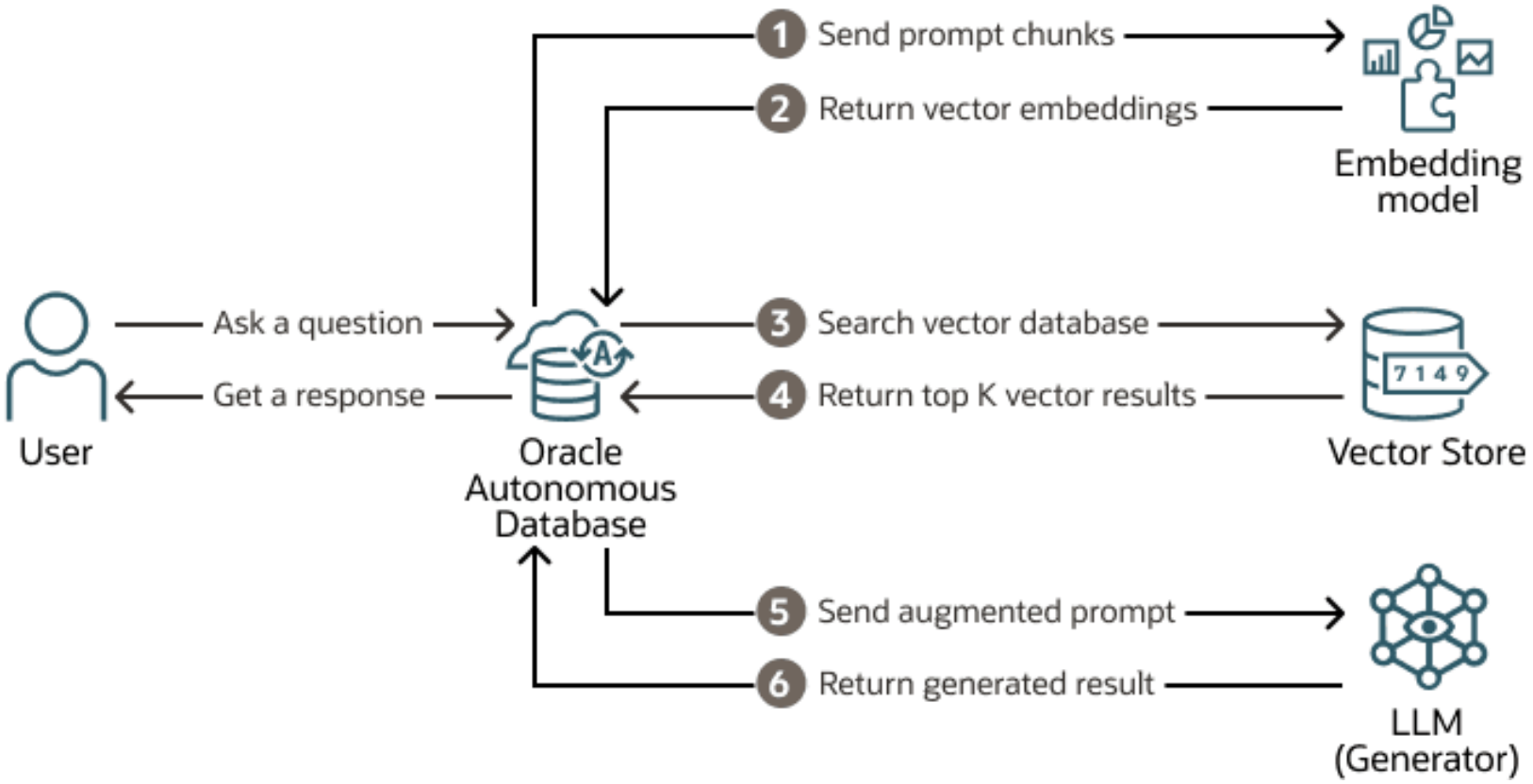
Similar input should result in similar embedding (vector) values.

We can calculate distance between vectors to find similarity.

RAG – preparation



RAG – querying



Let’s build a RAG “movie search system”

dataset

```
# ls plots/ | wc -l
    6627
# cat plots/The_Matrix_\(1999\).txt
Computer programmer Thomas Anderson  is
secretly a hacker known as "Neo". He is
restless, eager and driven to learn the
meaning of cryptic references to the
"Matrix" appearing on his computer. A
female hacker named Trinity  confirms that
a man named Morpheus  can help him.
however, three Agents, led by Agent Smith,
arrest Neo and attempt to prevent him from
collaborating with Morpheus...
```

dataset

```
# oci os object bulk-upload -bn anowicki-movies --src-dir movie_plots/
```

Accepted file formats: PDF, DOC, JSON, XML, or HTML

Preparation

```
BEGIN
  DBMS_CLOUD_AI.create_profile(
      'OCI_VECTOR_INDEX',
      '{"provider": "oci",
        "credential_name": "OCI_CRED_",
        "oci_compartment_id": "ocid1...",
         "vector_index_name": "MY_INDEX",
        "region" :"eu-frankfurt-1"
       }');
END;
/


EXEC DBMS_CLOUD_AI.SET_PROFILE('OCI_VECTOR_INDEX');
```

```
BEGIN
  DBMS_CLOUD_AI.CREATE_VECTOR_INDEX(
    index_name  => 'MY_INDEX',
    attributes  =>
     '{"vector_db_provider": "oracle",
     "location": "https://objectstorage...",
     "object_storage_credential_name": "OCI_CRED_",
     "profile_name": "OCI_VECTOR_INDEX",
     "vector_dimension": 1024,
     "vector_distance_metric": "cosine",
     "chunk_overlap":128,
     "chunk_size":1024
    }');
END;
/
```

DEMO

Querying

```
SQL> select ai narrate movies about hackers;

RESPONSE
--------------------------------------------------------------------------------
The context provides information about several movies related to hackers, including "Hackers (1995)",
"Live Free or Die Hard (2007)", and "The Net (1995)". These movies feature hackers as main characters and
involve plots related to hacking, cyber warfare, and computer security.
...
Sources:
  - Hackers_(1995).txt (https://..../Hackers_(1995).txt)
  - Live_Free_or_Die_Hard_(2007).txt (https://..../Live_Free_or_Die_Hard_(2007).txt)
  - The_Net_(1995).txt (https://.../The_Net_(1995).txt)

SQL> select ai narrate movies about a guy stuck in a canyon

RESPONSE
--------------------------------------------------------------------------------
The movie "127 Hours" is about a guy named Aron Ralston who gets stuck in a canyon. He is canyoneering in
Utah's Canyonlands National Park when a boulder falls on his arm, trapping him.
...
Sources:
  - 127_Hours_(2010).txt (https://..../127_Hours_(2010).txt)
```

Querying

```
SQL> select ai movies about a guy stuck in a canyon;

DATA                                    SOURCE                URL                                  SCORE START_OFFSET END_OFFSET
--------------------------------------- --------------------- ---------------------------------- ------ ------------ ----------
ey doubt he will show. Ralston continues 127_Hours_(2010).txt   https://...127_Hours_(2010).txt       .53          897       1920
n makes his way out of the canyon, where 127_Hours_(2010).txt   https://...127_Hours_(2010).txt       .47         2689       3402
lf out of water and is forced to drink h 127_Hours_(2010).txt   https://...127_Hours_(2010).txt       .47         1793       2816

SQL> select response from (select ai narrate movies about a guy stuck in a canyon) where rownum < 2;

select response from (select ai narrate movies about a guy stuck in a canyon) where rownum < 2
                                          *
ERROR at line 1:
ORA-00923: FROM keyword not found where expected
Help: https://docs.oracle.com/error-help/db/ora-00923/
```

What about real life usage?

What about real life usage?

Frequent questions:

- Can I stop it from exposing my data to the LLM?

Yes! `exec DBMS_CLOUD_AI.DISABLE_DATA_ACCESS;`

- Can I do it in an on-prem DB?

No, you need to wait for 26ai on-prem (unless you have Exadata running 23.7 or newer)

- Can I can do a self-hosted LLM service?

Yes, OpenAI-compatible providers are supported. Set a custom `provider_endpoint`

Only works if your endpoint uses 443 port for https

Select AI - summary

Querying data using natural language, conversations

- Generating SQL statements
- Explaining SQL statements

Synthetic data generation

RAG systems – Retrieval Augmented Generation

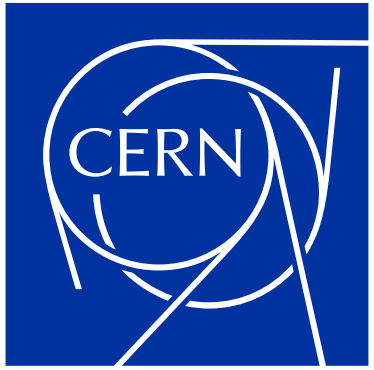
The feature set is constantly changing, we've only covered some part

References

Oracle Docs	https://docs.oracle.com/en/database/oracle/oracle-database/index.html
Oracle Blogs	https://blogs.oracle.com/database/
	https://blogs.oracle.com/machinelearning/post/supercharge-sql-unleashing-ai-queries-with-oracle-adb-select-ai
	https://docs.oracle.com/en/cloud/paas/autonomous-database/serverless/adbsb/whats-new-adwc.html
Movies dataset	https://www.cs.cmu.edu/~ark/personas/


If you’re ever in Geneva area…
https://visit.cern

Thank you !



andrzejnowicki

andrzej.nowicki@cern.ch

www.andrzejnowicki.pl

Slides are available: