idsia Dalle Molle Institute for Artificial Intelligence USI – SUPSI

Ask Your Data Anything!
LLMs for Effortless Database Insights

Swiss AI Weeks 2025

Fernando de Meer Pardo
&
Marco Pierbattista

USI SUPSI

Dalle Molle Institute for Artificial Intelligence USI – SUPSI

Speakers

Fernando de Meer Pardo, PhD

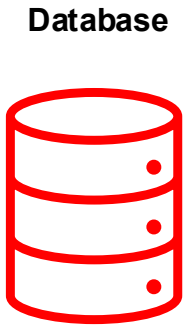
Marco Pierbattista, PhD

Any public names or entities mentioned in this presentation are used purely for hypothetical and illustrative purposes. They do not represent real-world analysis, endorsement, or affiliation and should not be relied upon for decision-making.

SUPSI

Data Access in RDBs
Current Status Quo

- Most users do not know how to write SQL queries
- Relational DBs can have **extremely complex schema** (tables & columns)
- Most data extractions are done via **hard-coded queries** (i.e. using a DB as a module of another application)
- In **exploratory tasks** (i.e background checks) users need to request data from DBAs who must spend time manually writing complex queries ==> **extracting information from DBs is slow and expensive**



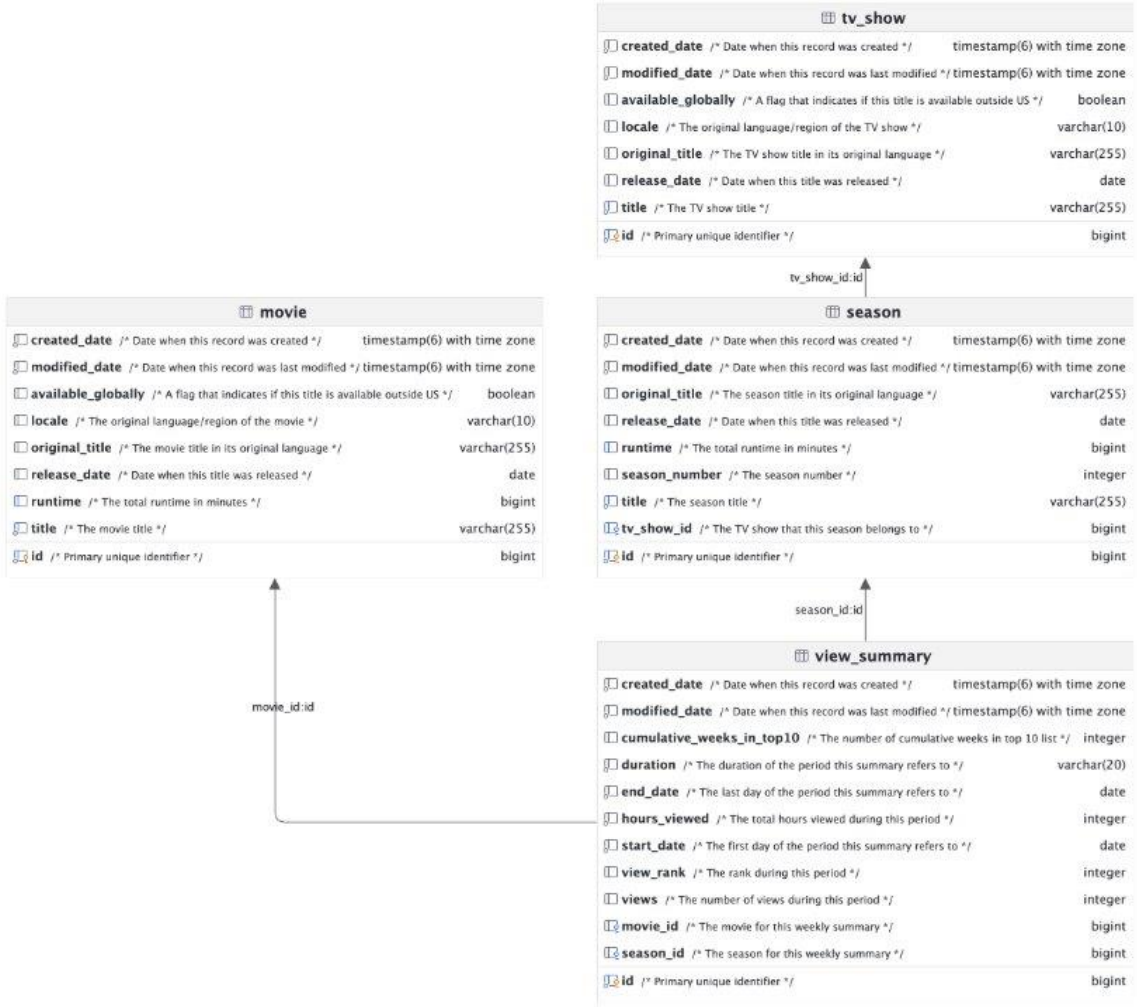
Goal: Allow users without any SQL knowledge to extract data from RDBs for exploratory tasks

Dalle Molle Institute for Artificial Intelligence USI – SUPSI

Data Access Running Example

- Open-source DB (available on Github) based on the data from the **Netflix Engagement Report** and the **Netflix Global Top 10 weekly list.**

- Simple schema yet exploratory tasks can lead to **complex queries**.



- The query below retrieves the **seasons of a given tv show** ranked by the number of views.

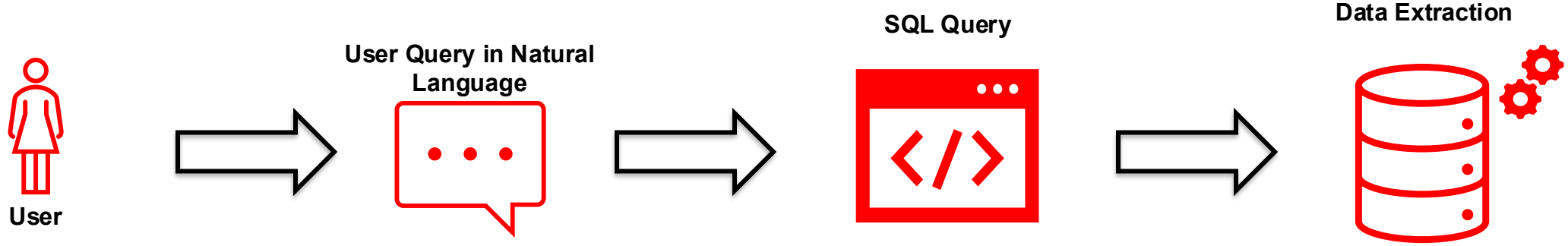
- It is a **relatively simple query**, but the multiple JOINs, the GROUP BY, ORDER BY and RANK() can be intimidating for users without SQL knowledge.

Example TV Show:

Floor is Lava


```sql
SELECT
    s.title AS season_title,
    s.season_number,
    SUM(vs.views) AS total_views,
    RANK() OVER (ORDER BY SUM(vs.views) DESC) AS view_rank
FROM
    tv_show tv
JOIN
    season s ON tv.id = s.tv_show_id
JOIN
    view_summary vs ON s.id = vs.season_id
WHERE
    tv.title = 'Floor is Lava'
GROUP BY
    s.id, s.title, s.season_number
ORDER BY
    total_views DESC;
```

NL2SQL overview



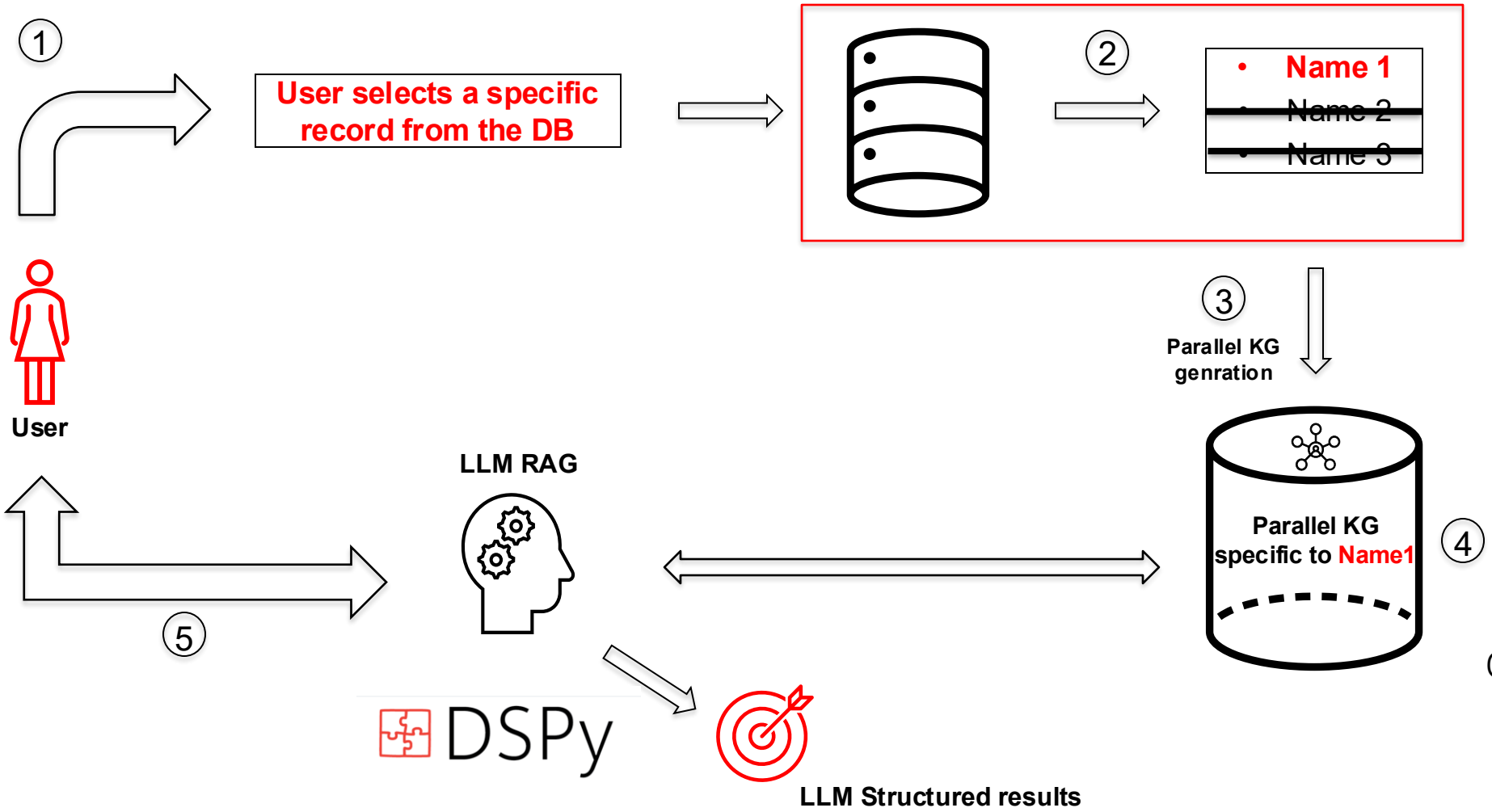
- **VERY COMPLEX** research problem in general (**difficult to evaluate** performance + easy to fail in multiple ways)
- NL queries and SQL queries have conflicting natures (**ambiguous** vs **strictly formatted**)
- Most deployed systems can only aid users (greatly reducing query writing time), but **some SQL knowledge from users is still required**

However, in our case, we are only interested in a reduced number of rows at a time…

Example TV Show:
Floor is Lava

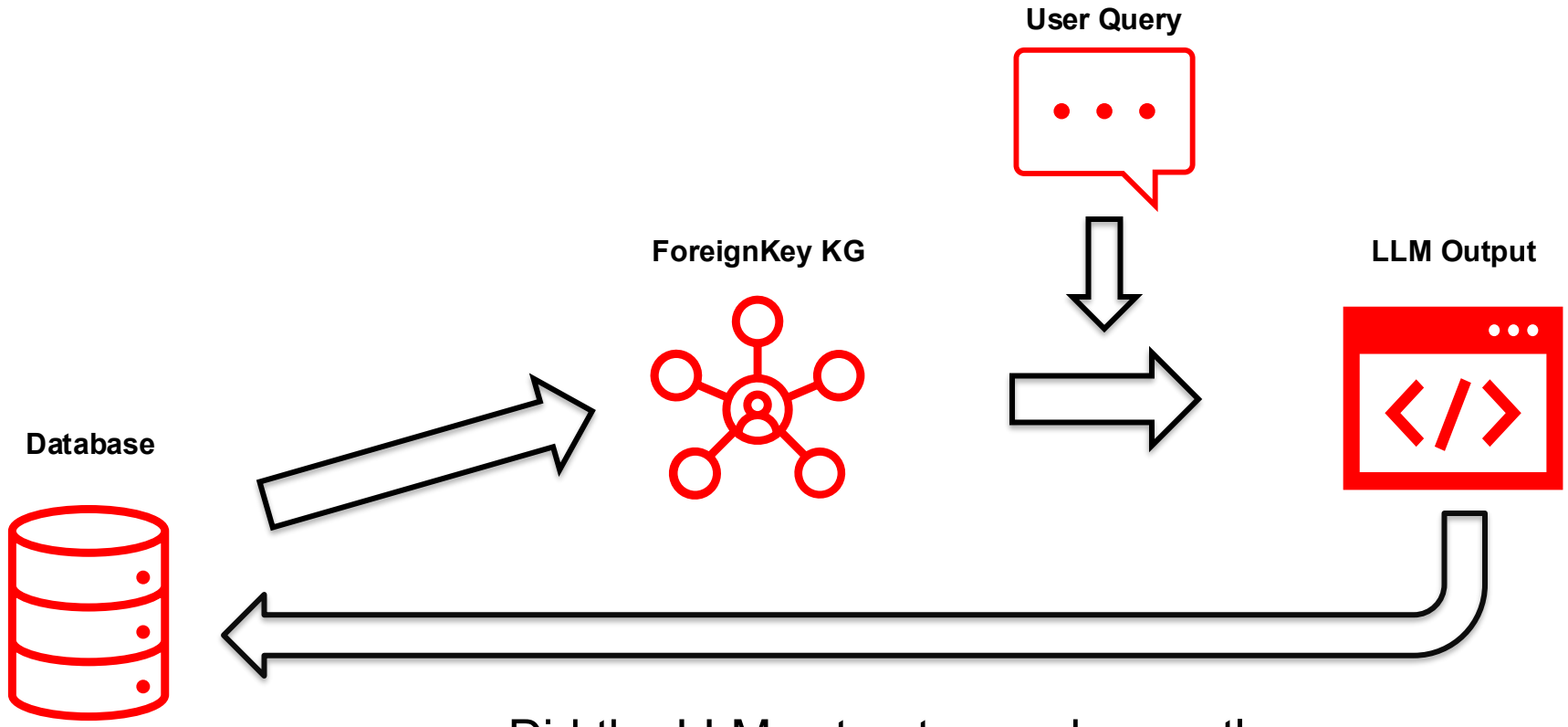
- All of the information related to a specific tv show can be accessed by exploring the **Primary/Foreign Key relationships** in the DB, which imply a Knowledge Graph.
- Users could manually explore the KG searching for the information they need, but this is **infeasible when dealing with complex schemas**.

Methodology



© IDSIA. All rights reserved.

Authenticity Guardrail



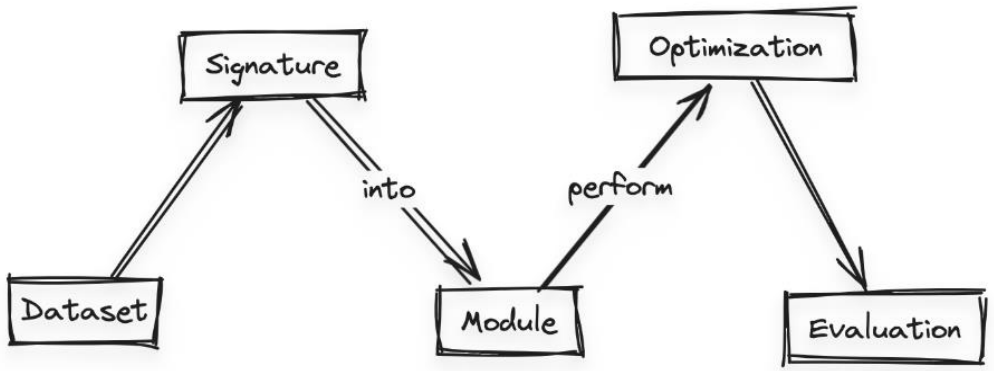
Did the LLM extract records exactly matching those in the DB?
We can **always** validate record extraction.

© IDSIA. All rights reserved.

Methodology

DSPy *Programming—not prompting—LMs*

DSPy is the framework for *programming—rather than prompting—language models*. It allows you to iterate fast on **building modular AI systems** and offers algorithms for **optimizing their prompts and weights**, whether you're building simple classifiers, sophisticated RAG pipelines, or Agent loops.

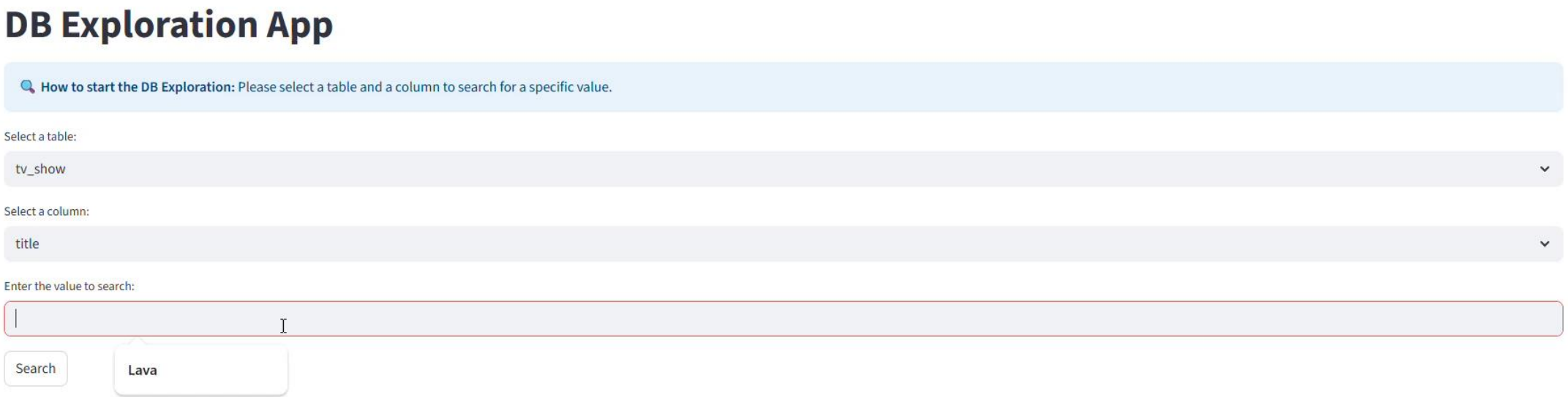



```python
class MyProgram(dspy.Module):

    def __init__(self, ...):
        # Define attributes and sub-modules here
        {constructor_code}

    def forward(self, input_name1, input_name2, ...):
        # Implement your program's logic here
        {custom_logic_code}
```

Dalle Molle Institute for Artificial Intelligence USI – SUPSI

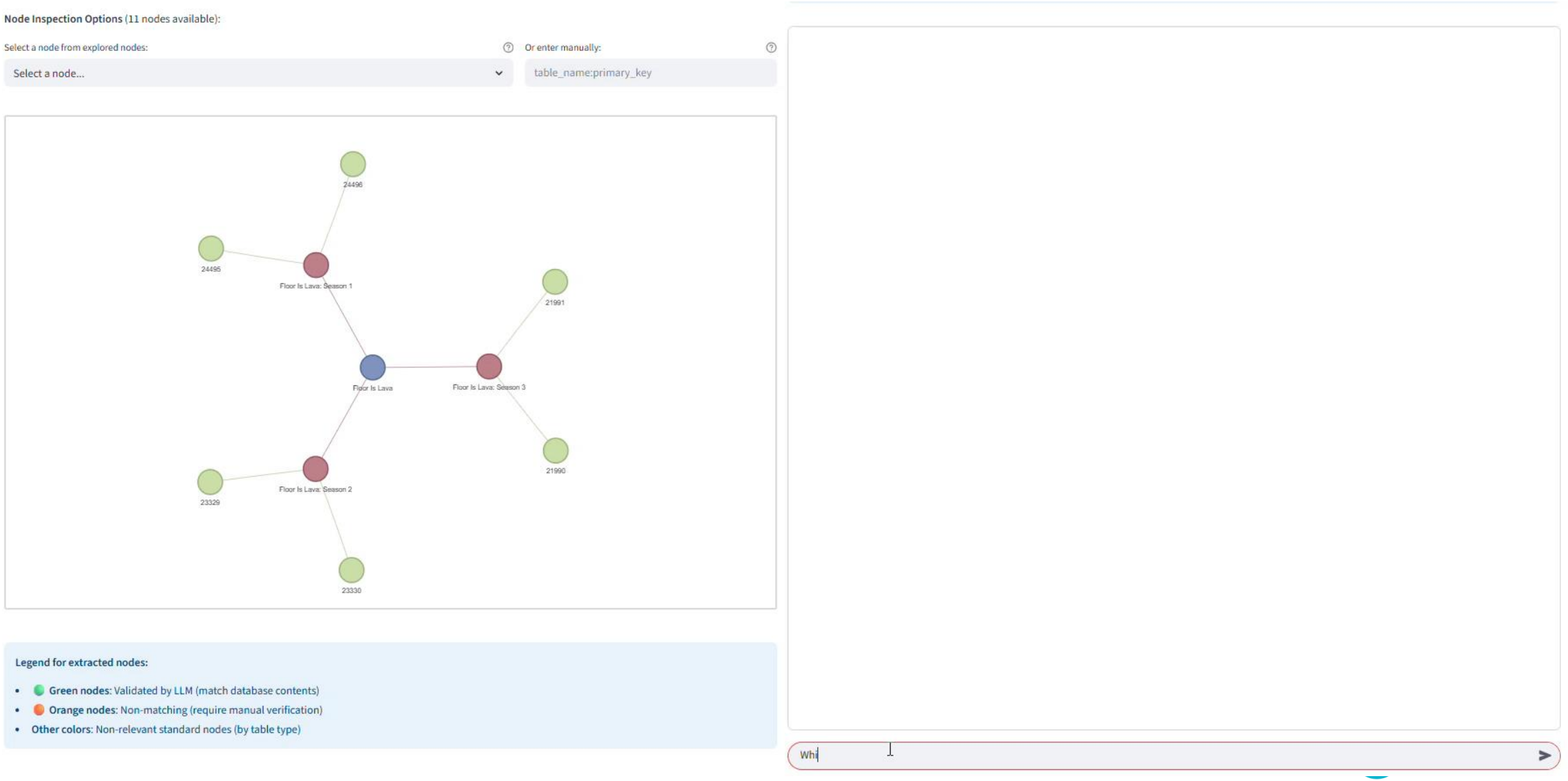
PoC video walkthrough

Deploy



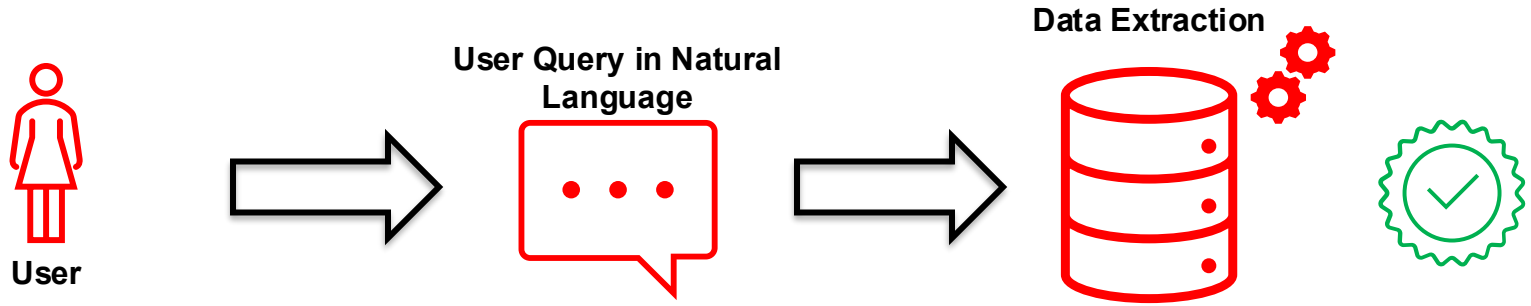
Dalle Molle Institute for Artificial Intelligence USI – SUPSI

Auth Guardrail video walkthrough



Conclusions

- NL2SQL is a very complex problem in general, but extracting information from small sections of a DB is a **much easier task**.
- Current LLMs are **very proficient** at simple, self-contained tasks.
- The rich structure of RDBs **allows for the verification** of extracted data.



idsia Dalle Molle Institute for Artificial Intelligence USI – SUPSI

{ai}

Demo / QA

Thank you for your attention!

SUPSI