LLM2026	**COSMIC-RAGSQL**		CC1
Coordinated by:	Mehdi Nechadi		2 months
Course : Theory and Practical Applications of Large Language Models			

COSMIC-RAGSQL: Cognitive Observation SysteM with RAG for SQL Queries

I Pre-proposal context, positioning and objective(s)

a Context

In contemporary Astronomy, the primary challenges are linked with the Big Data, modern observatories collect a lot of data that cannot be analyzed by humans themselves because of their size (often exceeding billions of objects). This makes SQL schemas hard to understand for beginners and even for researchers without any data understanding, which limits the democratization and accessibility of the knowledge within the educational and scientific community.

The integration of Artificial Intelligence has led to significant progress in data accessibility through Text-to-SQL applications. That architecture allows users to ask for concise information using Natural Language Processing without any strong knowledge of SQL.

However, this architecture is not sufficient to address all the challenges when it comes to observational astronomy. A user cannot know which celestial objects are visible for a location and a time using Text-to-SQL alone for two reasons:

- The SQL Database is static; there is no real-time position information in the database nor in the data online.
- The LLM cannot guarantee an absolute precision and determinism required for astronomic computations involving space and time constraints [1, 2].

This limitation highlights a crucial architectural gap and, consequently, the proposed solution must satisfy these four key requirements:

- Dynamic: The solution must integrate the space-time constraints and accurately convert dynamic observational needs into static database queries.
- Determinist: The architecture must delegate the calculation to ensure absolute accuracy for the constraint generation.
- Pedagogical: The system must deliver accessible and pedagogically enriched output to the user.
- Traceability: The system must provide complete auditability of the data source, linking the final output back to the specific SQL schema and RAG corpus used.

This project aims to directly address the limitation by implementing a system that can handle both complex computation and efficient SQL query generation to provide users with pedagogical information for spatial observation.

a1 Positioning to state of the art

With the evolution of AI and, more precisely, natural language processing, we have made significant changes in data understanding. The initial text-to-SQL applications were predicated on fine-tuning neural networks using a dataset of (Questions in natural language, Query in SQL) pairs. This allowed the network to learn the SQL schemas based on extensive training data. SQLNet [3] was the key model that performed well on the WikiSQL benchmark for several years.

However, these models required considerable data and were static, a single change in the SQL schemas would suffice to cause failures on every query. When strong LLM emerged with a large number of parameters, they addressed the limitation. By writing the SQL schemas directly in the prompt, it allows for more precise query generation. The most significant advancement was the Retrieval Augmented Generation (RAG) [4] which enables the LLM to get access to verified external data, increasing the quality of the output by limiting query errors. The RAG architecture also overcame the context-size and the database schema size limitations [5].

Nevertheless, the computational limitations of LLM remained unsolved until the Agentic AI paradigm emerged. Actually, an LLM with RAG cannot compute complex expressions because of its inherent statistical nature. Since the model operates by predicting the next token, the mathematical task led to error due to a non-deterministic output. The Agentic AI was developed to add reasoning and external behaviour to an LLM [6, 7]. It addresses the limitation above by allowing the LLM to delegate computation to an external deterministic function, thus entirely eliminating the risk of mathematical errors and hallucinations.

An existing architecture, StarWhisper Telescope [8] successfully combined the RAG and the Agentic methodologies to manage end-to-end spatial observations for the autonomous discovery of new celestial events.

However, the SWT is designed to help scientists and researchers, focusing on autonomous telescope operation and the detection of celestial events whereas our project aims to directly address the gap by creating a conversational agent which can provide high-quality, rich information about celestial objects and dynamic observation planning capabilities to users regardless of their degree of knowledge in Astronomy.

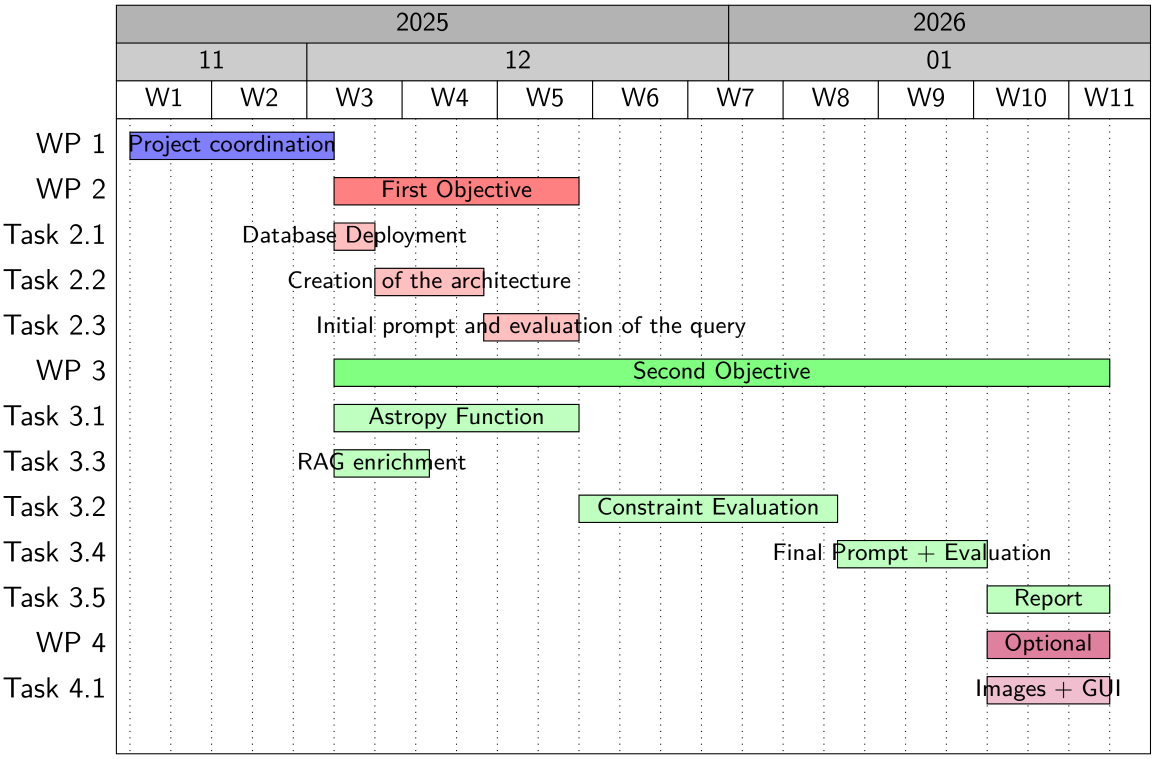
b Project's objectives and Methodology

b1 Objective 1: Architectural foundation and RAG Integration

Our first objective will be the design of our Architecture,we will use LangChain agentic framework and Gpt-4o for its reasoning capabilities. For the database, we will be using a license-free database named OpenNGC [9] that includes the entire NGC, IC and Messier catalog which we will deploy locally. We then add the RAG for the SQL schemas, and the SQL query tool. Following this, we will craft a prompt that guides our agent to use all the knowledge retrieved by the RAG (the schema) to generate the SQL query and pass it to the query executor tool. We will then evaluate this query afterwards.

b2 Objective 2: Deterministic Tool Integration

For this, we will create a deterministic function using the Astropy library to calculate the celestial position and derive the spatio-temporal parameters (e.g., coordinate boundaries). The Agent will then integrate these parameters to construct the final SQL constraint within the query. We will add more resources in the RAG in order to give a full description of each object and some subjective adjectives linked. Afterwards we can start the evaluation of the queries. If time permits eventually we will add the images of objects in a table to be able to return images of each object and a proper GUI to integrate our application.



References

[1] Berk Atil, Sarp Aykent, Alexa Chittams, Lisheng Fu, Rebecca J. Passonneau, Evan Radcliffe, Guru Rajan Rajagopal, Adam Sloan, Tomasz Tudrej, Ferhan Ture, Zhe Wu, Lixinyu Xu, and Breck Baldwin. *Non-Determinism of "Deterministic" LLM Settings*. 2025. arXiv: 2408.04667 [cs.CL]. URL: https://arxiv.org/abs/2408.04667.

[2] Johan Boye and Birger Moell. *Large Language Models and Mathematical Reasoning Failures*. 2025. arXiv: 2502.11574 [cs.AI]. URL: https://arxiv.org/abs/2502.11574.

[3] Xiaojun Xu, Chang Liu, and Dawn Song. *SQLNet: Generating Structured Queries From Natural Language Without Reinforcement Learning*. 2017. arXiv: 1711.04436 [cs.CL]. URL: https://arxiv.org/abs/1711.04436.

[4] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*. 2021. arXiv: 2005.11401 [cs.CL]. URL: https://arxiv.org/abs/2005.11401.

[5] Prakhar Gurawa and Anjali Dharmik. *Balancing Content Size in RAG-Text2SQL System*. 2025. arXiv: 2502.15723 [cs.IR]. URL: https://arxiv.org/abs/2502.15723.

[6] Antony Seabra, Claudio Cavalcante, Joao Nepomuceno, Lucas Lago, Nicolaas Ruberg, and Sergio Lifschitz. *Contrato360 2.0: A Document and Database-Driven Question-Answer System using Large Language Models and Agents*. 2024. arXiv: 2412.17942 [cs.AI]. URL: https://arxiv.org/abs/2412.17942.

[7] Changle Qu, Sunhao Dai, Xiaochi Wei, Hengyi Cai, Shuaiqiang Wang, Dawei Yin, Jun Xu, and Ji-rong Wen. “Tool learning with large language models: a survey”. In: *Frontiers of Computer Science* 19.8 (Jan. 2025). ISSN: 2095-2236. DOI: 10.1007/s11704-024-40678-2. URL: http://dx.doi.org/10.1007/s11704-024-40678-2.

[8] Cunshi Wang, Yu Zhang, Yuyang Li, Xinjie Hu, Yiming Mao, Xunhao Chen, Pengliang Du, Rui Wang, Ying Wu, Hang Yang, Yansong Li, Beichuan Wang, Haiyang Mu, Zheng Wang, Jianfeng Tian, Liang Ge, Yongna Mao, Shengming Li, Xiaomeng Lu, Jinhang Zou, Yang Huang, Ningchen Sun, Jie Zheng, Min He, Yu Bai, Junjie Jin, Hong Wu, and Jifeng Liu. *StarWhisper Telescope: An AI framework for automating end-to-end astronomical observations*. 2025. arXiv: 2412.06412 [astro-ph.IM]. URL: https://arxiv.org/abs/2412.06412.

[9] Mattia Verga. *OpenNGC: A license friendly NGC/IC objects database*. GitHub. 2023. URL: https://github.com/mattiaverga/OpenNGC.