Digital Rules:
Graph_RAG vs RAG Case Study

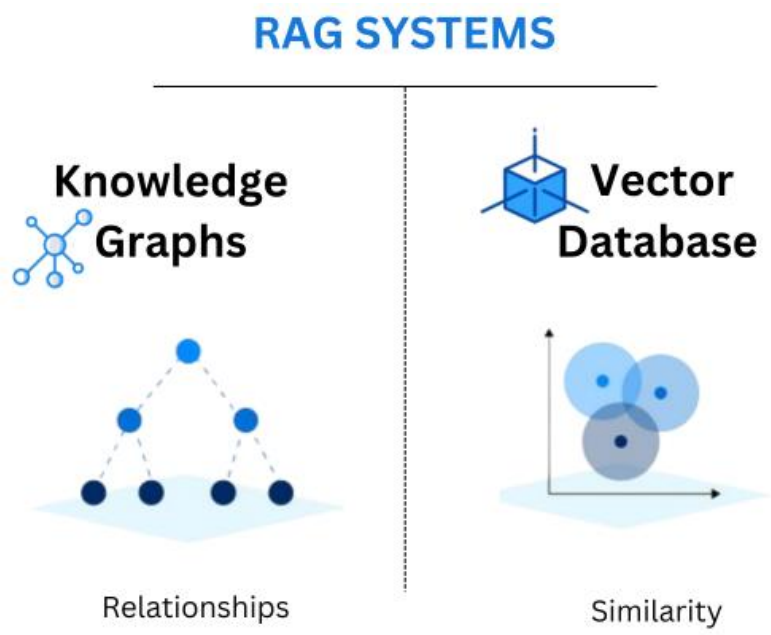
Leizheng Wang

Background and Objective: RAG vs GraphRAG

Objective:

- A high-performance RAG chatbot getting precise and concise answer from complex documentation.
- Test the pro and cons for both systems.

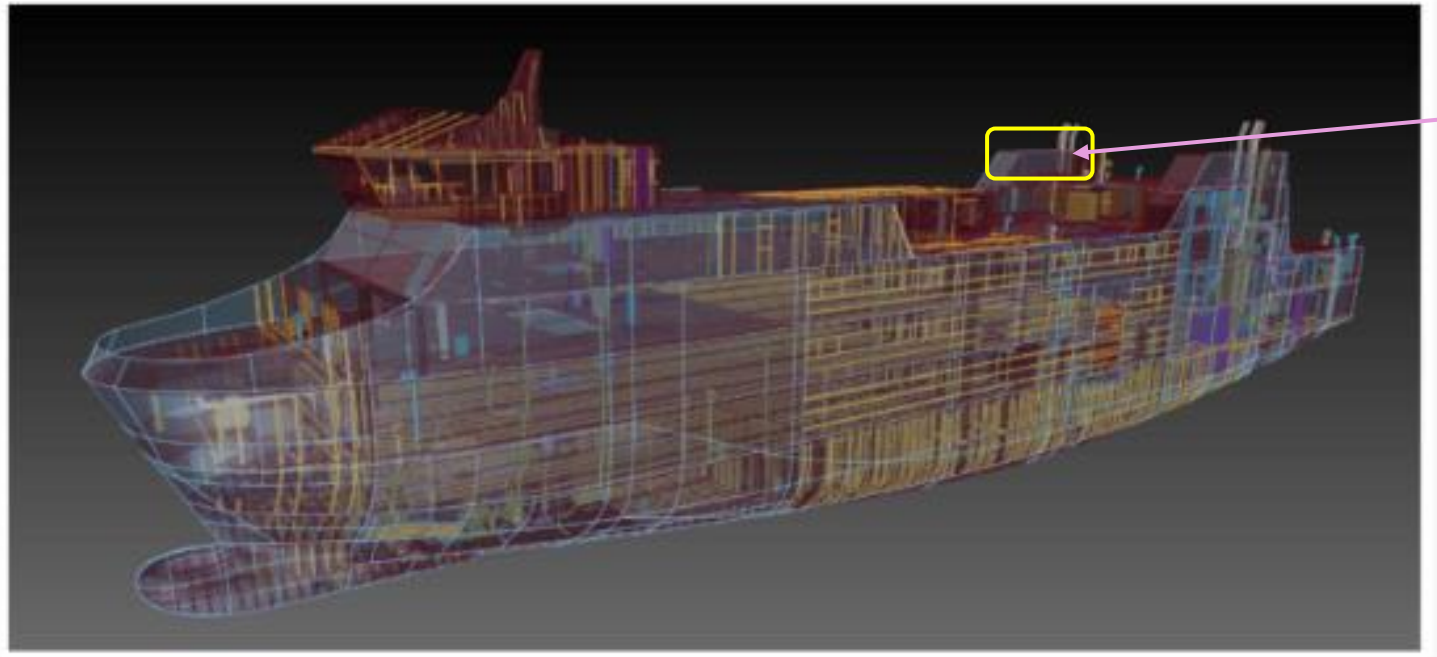
Perspective	Vector RAG	GraphRAG
Data Structure	Vector Database	Knowledge Graph
Focus	Similarity seach	Entities, relationships and contect
Deep Understanding	deeper insights throgh entity connections	Deeper insdights through entity connection and semantic structures
Retrival Speed	Generaly faster for large dataset	maight be slower for complex queries
Scalability	highly scalable for unstructured data	can be less saclable for very large knowledge graph
Cost	Relatively cheap	Expensive for training and hardware
Workload	Simple	Need to prepare detail schema, the more precise schedma, the more accurate results
Application	simple text	Industrial production level, need high precision and concise results for complicated task



Application scenario: ship construction design

- A leading company specializing in making world-wide ship construction and survey rules.
- Rules apply for different ship, boat and marine vessels, offshore rig.
- from system (engine, propeller, shaft), safety regulation to small components (bolts, screw…).

❖ Goal: a chat bot accurately answers detail design rule and safety requirements for any part of a ship.



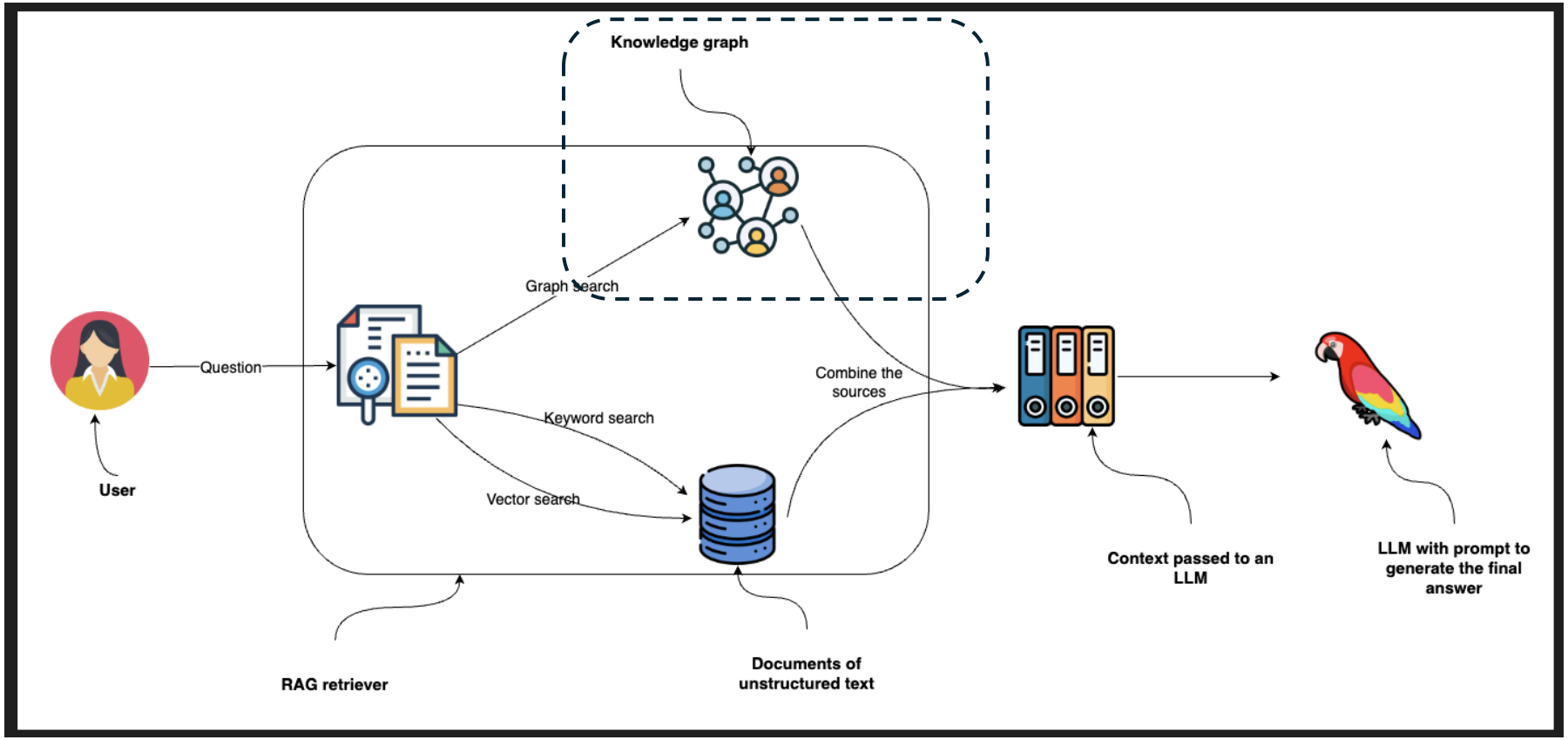
1:Design parameters:
Value range (-)

2: Formula:

3: Safety requirement

4.:

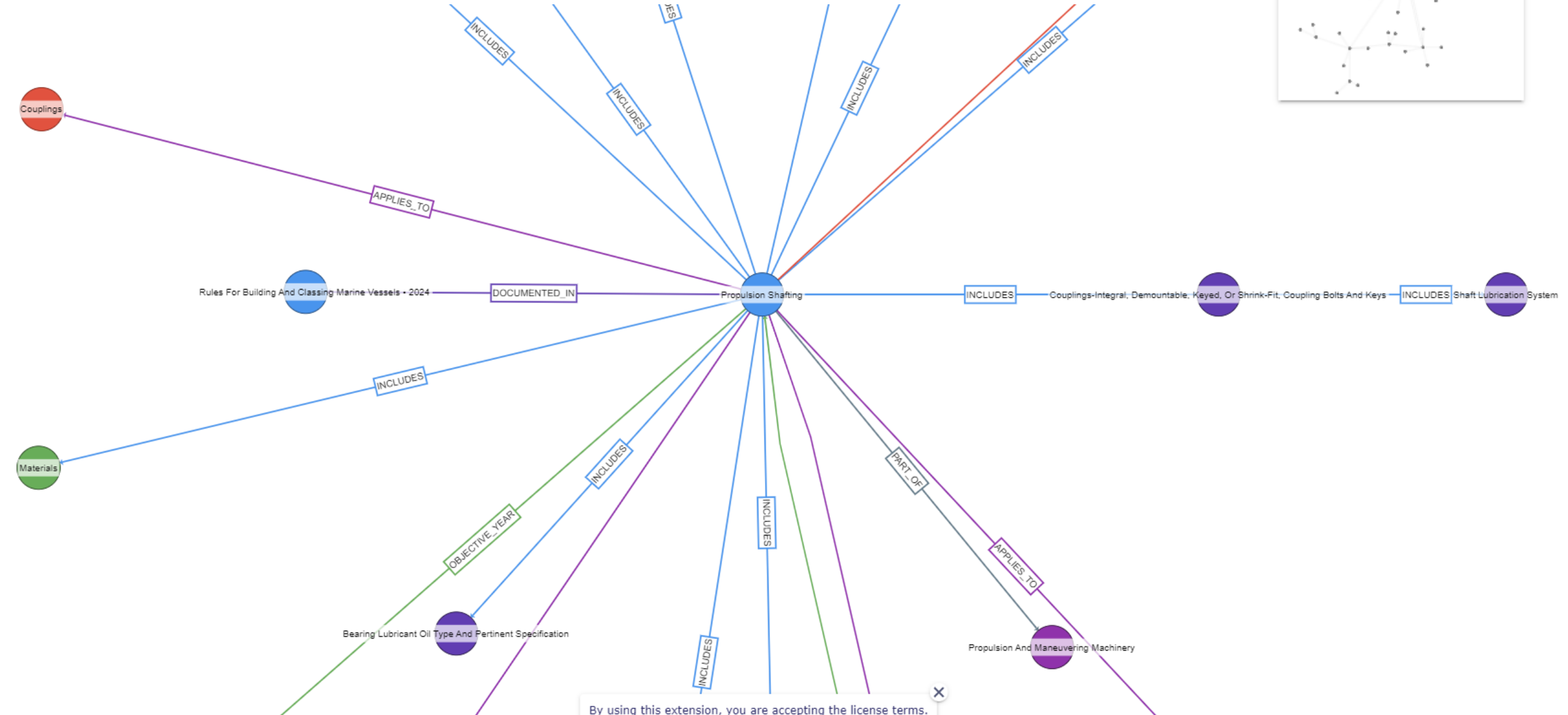
Methodology: Hybrid Retrieval

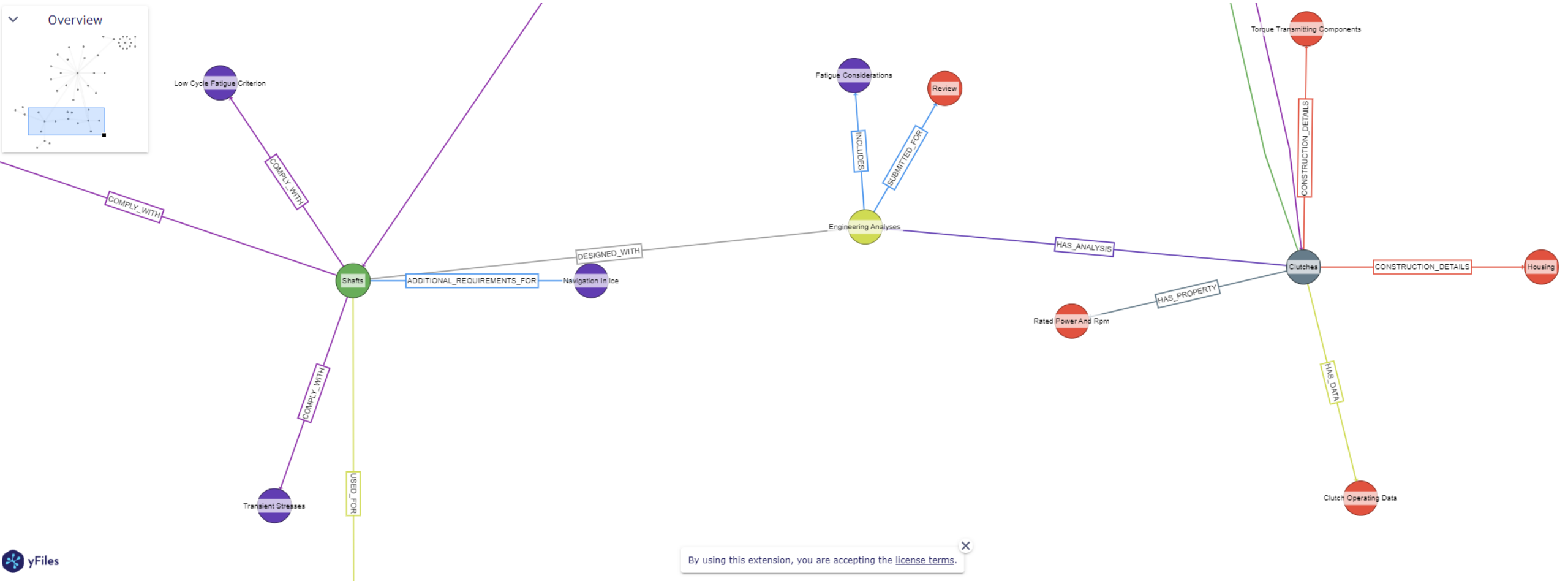


- Combine the content from both graph retrieval and vector retrieval

```
chain = (
    RunnableParallel(
        {
            "context": _search_query | retriever,
            "question": RunnablePassthrough(),
        }
    )
    | prompt
    | llm
    | StrOutputParser()
)
```

Extracted graph example built in neo4j




Overview
Low Cycle Fatigue Criterion
COMPLY_WITH
COMPLY_WITH
Shafts
ADDITIONAL_REQUIREMENTS_FOR
Navigation In Ice
DESIGNED_WITH
COMPLY_WITH
Transient Stresses
USED_FOR
Fatigue Considerations
Review
INCLUDES
SUBMITTED_FOR
Engineering Analyses
HAS_ANALYSIS
Torque Transmitting Components
CONSTRUCTION_DETAILS
Clutches
CONSTRUCTION_DETAILS
Housing
HAS_PROPERTY
Rated Power And Rpm
HAS_DATA
Clutch Operating Data
yFiles
By using this extension, you are accepting the license terms.

Results example 1:

RAG

Human: summarize this documentation

The documentation outlines requirements for mounting, securing, and electrical bonding arrangements, which need approval. Valves must be accessible and controllable, with indicators for open or closed states, and manual operation must be possible if power fails. Tank vents and sounding must comply with specific sections. For the TCM-PS notation, additional system requirements include having at least two independent seals with a buffer zone to reduce stress concentrations. The sterntube seal must be an approved type, having undergone design assessment and testing. Materials used must meet design requirements for the application, operating conditions, and environment, supporting Tier 1 goals. The design, construction, installation, and maintenance must meet functional requirements.

- 空洞
- 繁琐，

GraphRAG

'The documentation outlines the requirements and procedures for vessels with water-lubricated bearings, focusing on compliance with specific standards. Key aspects include the need for a stern inspection chamber, installation of a split type aftmost bearing with appropriate seals, and a seawater cooling system with redundancy. It also covers the maintenance of records for oil condition and replacement, anti-freeze properties of lubricant water, and alignment verification in the presence of a Surveyor. Additionally, it specifies the run-in procedure for certain shaft installations.

- 简洁，直达关键词

Results example 2:

RAG

Human: How many kinds of bolts are there, and what are the specific definitions and design concerns for each type?

> Finished chain.

> Finished chain.
The provided context does not specify different kinds of bolts or their specific definitions and design concerns. It does mention requirements for non-fitted bolts, such as ensuring the tensile stress does not exceed 90% of the minimum specified yield strength and providing means to prevent loosening in service. However, it does not provide a comprehensive list or detailed definitions for different types of bolts. If you need more detailed information, you might need to refer to a specific engineering or manufacturing standard that categorizes and defines bolt types.

GraphRAG

There are two kinds of bolts mentioned: coupling bolts and non-fitted bolts. \n\n1. **Coupling Bolts**: These can be integral, demountable, keyed, or shrink-fit. Specific details regarding the interference fit of the coupling bolts must be submitted, along with calculations and design basis for sizing if not based on the as-built line shaft diameter.\n\n2. **Non-fitted Bolts**: The tensile stress due to pre-stressing and astern pull should not exceed 90% of the minimum specified yield strength of the bolt material. Bearing stress on any member like the flange, bolt head, threads, or nut should also not exceed 90% of the yield strength of the material. These bolts must have means to prevent loosening in service.'

Results example 3: SPE 5 papers

RAG: vs GraphRAG:

Search query: how to use machine learning to improve unconventional production?

Machine learning can be used to improve unconventional production by creating a flexible methodology that generates production profiles for new shale wells. This methodology considers both local production information and general basin trends. The process involves four main steps:

1. Areal two-dimensional (2D) geological mapping based on data from vertical pilot wells.
2. Training machine learning models that connect varying geological properties and horizontal well completion parameters with production profiles.
3. Generating the production profiles for each cell of the 2D areal geological maps based on geological properties and optimal completion parameters.
4. Merging machine learning results with the production information from existing wells through cokriging.

This approach results in a map of play that highlights the most probable total gas production for different areas and the risk associated with that. The map can be used to rank locations for new wells for drilling. This methodology provides a quick and efficient first estimate, but a more
detailed data investigation is required to sanction a new well for a particular location.

- 繁琐
- 字面浅显意思，单纯copy只言片语

'To improve unconventional production using machine learning, a flexible, scalable, and integrated framework can be developed to leverage all sources of data. This involves transforming various data types with fit-for-purpose methodologies to evaluate and predict well performance. Machine learning can identify geology sweet spots, understand performance drivers, and optimize development strategies such as well spacing, completion, and production designs. By integrating machine learning with local decline curve data, it is possible to consider both global trends from the entire basin and local trends from nearby wells, leading to more realistic production forecasts and reduced uncertainty.

- Concise
- 深层理解(can generalize)

Conclusions:

1. GraphRAG exceeds far more than traditional RAG from both overall understanding and detail precision perspective.
2. GraphRAG is expensive to implement and train (>10 times).
3. GraphRAG highly depends on the quality of graph schema (bottleneck).
4. GraphRAG maybe have small scalability.
5. Current RAG: copying some sentences from side-perspectives, GPT-4o probably lacks in summarizing, reasoning and logic thinking capability.

Init

```python
# Initialize index
index_name = 'langchain-retrieval-augmentation-fast'
if index_name not in pc.list_indexes().names():
    pc.create_index(
        name=index_name,
        dimension=1536,
        metric='cosine',
        spec={
            "serverless": {
                "cloud": "aws",
                "region": "us-east-1"
            }
        }
    )
```

```python
# Chat function with streaming and DB storage
def call_chat(question):
    result = conversation.invoke({"question": question})

    answer = result["answer"]
    sources = result.get("source_documents", [])

    # Add source information to the answer
    source_info = "\n\nSources:"
    for doc in sources:
        source = doc.metadata.get('source', 'Unknown')
        page = doc.metadata.get('page', 'unknown')
        source_info += f"\n- {source}, page {page}"

    full_response = answer + source_info
```