Banks managing vast document repositories (e.g., mortgage loan packages) often consider using LLMs to interact with data. However, combining LLMs with embeddings and knowledge graphs (KGs) delivers significantly more power, reliability, and scalability than relying on LLMs alone.

✅ 1. LLMs Are Great at Reasoning — Not at Retrieval

- **LLMs lack memory** and context window limits restrict their ability to "see" large content sets.
- **Loan packages exceed prompt limits** and contain unstructured, multi-format data.

✉ **Solution**: Pre-indexing with **embeddings** enables **semantic search** and fast, accurate retrieval without needing to load entire documents into the prompt.

✅ 2. Knowledge Graphs Add Context, Trust & Traceability

- LLMs hallucinate and can't inherently understand how entities relate.
- Mortgage workflows require structured, explainable paths between entities (e.g., borrower, loan, document, officer).

✉ **Solution**: **Knowledge Graphs** define verified relationships and provide **filtering, compliance, and audit trail support**.

✅ 3. Multi-Modal Understanding Requires Specialized Embeddings

- Loan packages include **text, tables, scanned images, and layouted forms**.
- LLMs struggle to accurately parse and recall structured/tabular/image data.

✉ **Solution**: Use **modality-specific embeddings** (for text, tables, images, layout) to enable accurate cross-format search and retrieval.

✅ 4. Faster, Cheaper, Governable

- LLM queries are expensive; embeddings/KG are precomputed and fast.

- Embeddings allow **governed access**, redaction, and control over what data LLM sees.

✉ **Benefit**: **Low-latency, explainable AI** workflows.

✅ 5. Composable, Explainable Architecture

- KG + embeddings provide tools for **orchestration**, **reasoning**, and **fact validation**.
- Enables **dynamic workflows**, e.g., KG filtering → vector search → LLM summarization.

✅ 6. Long-Term Memory Across Sessions

- LLMs are **stateless**. They forget user actions after a session ends.
- Loan processors often revisit complex files over days/weeks.

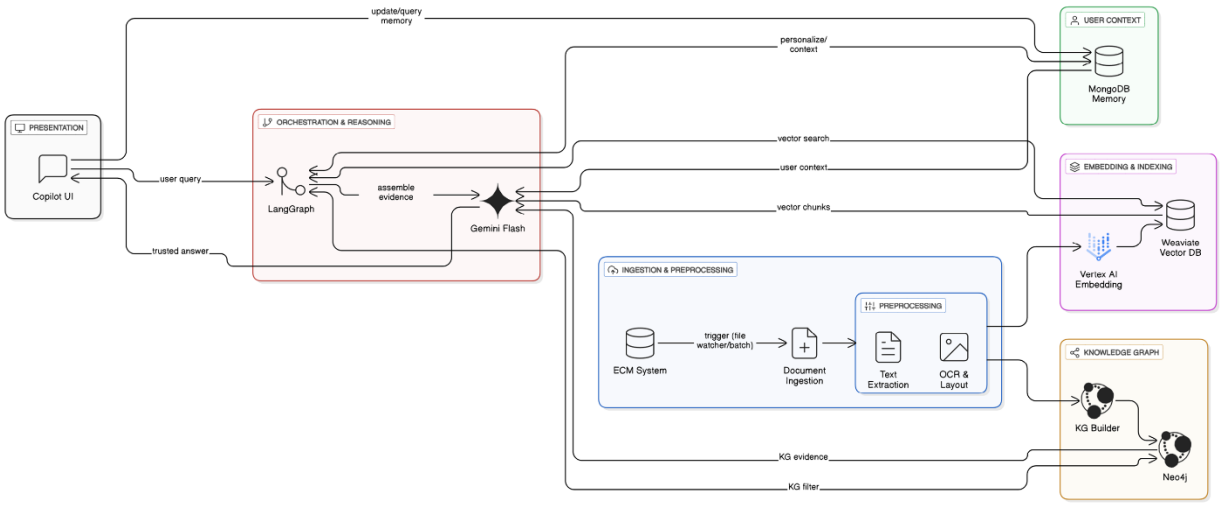
✉ **Solution**:

- Store **user query context, viewed documents, entities** as embeddings + KG nodes.
- On next login, **restore workflow state** and **personalize retrieval** based on past actions.

E.g., "Find where I left off in Loan #123's appraisal section."

🚀 Architecture Summary

Layer	Role
LLM	Reasoning, summarization, interaction
Vector DB	Fast semantic retrieval (text, image, table)
Knowledge Graph	Structured, explainable entity relationships
Embedding Memory	Long-term user memory, cross-session recall


update/query memory
personalize/ context
USER CONTEXT
MongoDB Memory
PRESENTATION
Copilot UI
ORCHESTRATION & REASONING
LangGraph
assemble evidence
Gemini Flash
vector search
user context
vector chunks
user query
trusted answer
EMBEDDING & INDEXING
Weaviate Vector DB
Vertex AI Embedding
INGESTION & PREPROCESSING
ECM System
trigger (file watcher/batch)
Document Ingestion
PREPROCESSING
Text Extraction
OCR & Layout
KNOWLEDGE GRAPH
KG Builder
Neo4j
KG evidence
KG filter