**UAE Legal GraphRAG — Reference Index**

**1) UAE legal context**

*(Background on the UAE legal system for domain grounding.)* ([nyulawglobal.org](https://www.nyulawglobal.org/globalex/united_arab_emirates1.html?utm_source=chatgpt.com))

* NYU GlobaLex — Overview of the United Arab Emirates legal system  
  <https://www.nyulawglobal.org/globalex/united_arab_emirates1.html>

**2) Microsoft GraphRAG — project & docs**

*(Official project, docs site, and core conceptual pages.)* ([GitHub](https://github.com/microsoft/graphrag?utm_source=chatgpt.com), [microsoft.github.io](https://microsoft.github.io/graphrag/?utm_source=chatgpt.com))

* GitHub (microsoft/graphrag) — source code  
  <https://github.com/microsoft/graphrag>
* Docs home (overview of the GraphRAG process)  
  <https://microsoft.github.io/graphrag/>
* Responsible AI / “What is GraphRAG?” (RAI transparency)  
  <https://github.com/microsoft/graphrag/blob/main/RAI_TRANSPARENCY.md#what-is-graphrag>
* Get started (quickstart)  
  <https://microsoft.github.io/graphrag/get_started/>
* Index / Overview  
  <https://microsoft.github.io/graphrag/index/overview/>
* Query overview  
  <https://microsoft.github.io/graphrag/query/overview/>
* Example notebook: global search  
  <https://microsoft.github.io/graphrag/examples_notebooks/global_search/>

**3) Neo4j GraphRAG for Python — official package, docs & examples**

*(Neo4j’s first-party GraphRAG package, guides, and examples.)* ([Graph Database & Analytics](https://neo4j.com/docs/neo4j-graphrag-python/current/?utm_source=chatgpt.com" \o "GraphRAG for Python))

* Docs index — **GraphRAG for Python**  
  <https://neo4j.com/docs/neo4j-graphrag-python/current/index.html>
* User guide: **KG builder** (pipeline structure)  
  <https://neo4j.com/docs/neo4j-graphrag-python/current/user_guide_kg_builder.html#pipeline-structure>
* User guide: **RAG configuration**  
  <https://neo4j.com/docs/neo4j-graphrag-python/current/user_guide_rag.html#graphrag-configuration>
* API reference: **Entity/Relation extraction template**  
  <https://neo4j.com/docs/neo4j-graphrag-python/current/api.html#erextractiontemplate>
* GitHub (neo4j/neo4j-graphrag-python) — source code  
  <https://github.com/neo4j/neo4j-graphrag-python>
* Examples repo (end-to-end notebooks)  
  <https://github.com/neo4j-product-examples/graphrag-python-examples/tree/main>
* Example notebook (End-to-End Lupus)  
  <https://github.com/neo4j-product-examples/graphrag-python-examples/blob/main/end-to-end-lupus.ipynb>
* Neo4j blog (announcement: GraphRAG Python package)  
  <https://neo4j.com/blog/news/graphrag-python-package/>

**4) Neo4j ecosystem, products & use-cases**

*(Products and background articles you’ll cite in architecture/cost slides.)* ([Graph Database & Analytics](https://neo4j.com/product/auradb/?utm_source=chatgpt.com" \o "Neo4j AuraDB: Fully Managed Graph Database))

* AuraDB (managed Neo4j) — product page  
  <https://neo4j.com/product/auradb/?ref=neo4j-home-hero>
* Bloom (visual exploration)  
  <https://neo4j.com/product/bloom/>
* Generative AI hub (Neo4j + GenAI resources)  
  <https://neo4j.com/generativeai/>
* GenAI ecosystem (Neo4j Labs)  
  <https://neo4j.com/labs/genai-ecosystem/>
* Developer use case: AI for customer experiences  
  <https://neo4j.com/developer/genai-ecosystem/ai-for-customer-experiences/>
* Use cases: knowledge graph  
  <https://neo4j.com/use-cases/knowledge-graph/>
* Blog: **What is RAG?**  
  <https://neo4j.com/blog/genai/what-is-retrieval-augmented-generation-rag/>
* Blog: **What is GraphRAG?**  
  <https://neo4j.com/blog/genai/what-is-graphrag/>
* Blog: **What is generative AI?**  
  <https://neo4j.com/blog/genai/what-is-generative-ai/>
* Blog: **GraphRAG manifesto**  
  <https://neo4j.com/blog/genai/graphrag-manifesto/>
* News: **Neo4j × Microsoft collaboration**  
  <https://neo4j.com/blog/news/neo4j-microsoft-collaboration/>
* Neo4j Spark connector — **Azure Synapse Analytics** guide  
  <https://neo4j.com/docs/spark/current/data-warehouses/azure-synapse-analytics/>
* Azure reference deployment (Neo4j + generative AI)  
  <https://github.com/neo4j-partners/neo4j-generative-ai-azure>

**5) Knowledge-graph foundations & industry context**

*(Foundational perspective pieces to cite in introductions/appendix.)* ([blog.google](https://blog.google/products/search/introducing-knowledge-graph-things-not/?utm_source=chatgpt.com" \o "Introducing the Knowledge Graph: things, not strings))

* Google’s announcement: **Knowledge Graph — things, not strings** (2012)  
  <https://blog.google/products/search/introducing-knowledge-graph-things-not/>
* Neo4j blog: **What is a knowledge graph?**  
  <https://neo4j.com/blog/genai/what-is-knowledge-graph/>
* Medium (enterprise focus): **Implementing knowledge graphs + LLMs**  
  <https://medium.com/data-science/how-to-implement-knowledge-graphs-and-large-language-models-llms-together-at-the-enterprise-level-cf2835475c47>
* Medium (MongoDB): **Knowledge-graph RAG using MongoDB**  
  <https://medium.com/mongodb/knowledge-graph-rag-using-mongodb-1346e953064c>
* Blog essay: **GNNs meet LLMs** (survey/notes)  
  <https://n1o.github.io/posts/graph-neural-networks-meet-large-language-models/>
* GitHub repo (curation): **kg\_llm**  
  <https://github.com/SteveHedden/kg_llm>

**6) Cloud & graph interop (Azure)**

*(Gremlin API and graph integration references.)* ([Microsoft Learn](https://learn.microsoft.com/en-us/azure/cosmos-db/gremlin/overview?utm_source=chatgpt.com))

* Azure Cosmos DB — **Gremlin/Apache TinkerPop** overview  
  <https://learn.microsoft.com/en-us/azure/cosmos-db/gremlin/overview>

**7) Research, tooling & media**

*(Supplementary tools, research, and media items for evaluation or inspiration.)* ([GitHub](https://github.com/graspologic-org/graspologic?utm_source=chatgpt.com), [NVIDIA Developer](https://developer.nvidia.com/blog/boosting-qa-accuracy-with-graphrag-using-pyg-and-graph-databases/?utm_source=chatgpt.com))

* graspologic — graph analytics for Python  
  <https://github.com/graspologic-org/graspologic>
* NVIDIA technical blog: **Boosting Q&A accuracy with GraphRAG (PyG + graph DBs)**  
  <https://developer.nvidia.com/blog/boosting-qa-accuracy-with-graphrag-using-pyg-and-graph-databases/>
* arXiv paper (as provided)  
  <https://arxiv.org/abs/2503.21322>
* Infranodus (network text analysis) — pricing page  
  <https://infranodus.com/#pricing>
* YouTube video (as provided)  
  <https://www.youtube.com/watch?v=GgOT2_cm7ZY>

**8) Additional GraphRAG reference (independent site)**

*(Memory/knowledge-graph reference page that complements the MS docs.)*

* graphrag.com — **Memory graph reference**  
  <https://graphrag.com/reference/knowledge-graph/memory-graph/>

**9) (Optional) Microsoft Research write-up**

*(Blog/press item announcing GraphRAG on GitHub.)* ([Microsoft](https://www.microsoft.com/en-us/research/blog/graphrag-new-tool-for-complex-data-discovery-now-on-github/?utm_source=chatgpt.com))

* Microsoft Research blog — **GraphRAG now on GitHub**  
  <https://www.microsoft.com/en-us/research/blog/graphrag-new-tool-for-complex-data-discovery-now-on-github/>