Knowledge Graph for RAG — AWS & OSS Swimlane (One Pager)

Data → KG → Serving | CQs, RDF/RDFS/OWL, SKOS, SHACL, SPARQL, Federated SPARQL

**YOUR LOGO**

# [0] Define Competency Questions (CQs)

Purpose: Scope answers; drives ontology & ETL.  
Tools: AWS QuickSight; OSS Markdown/ADR in Git, Confluence/Notion.

|  |  |  |
| --- | --- | --- |
| DATA | KG | SERVING |
| **[1] Land raw sources → Bronze**  • Inputs: Sales DB, R&D PDFs, OCR PDFs, CAD/Images, XML/JSON, Reviews, Parts, Patents.  • AWS: S3, Kinesis, DMS, Glue Crawlers, Textract, AppFlow, KMS/IAM.  • OSS: Airbyte/NiFi, Tika (PDF), Tesseract (OCR), Scrapy/Playwright.  **[2] Clean & normalize → Silver**  • Parse CSV/JSON/XML, normalize dates/IDs, QA checks, light entity cues.  • AWS: Glue ETL (Spark), EMR Serverless, MWAA (Airflow), Great Expectations.  • OSS: Apache Spark, dbt-core (tests), Great Expectations, Airflow.  **[3] Conform → Gold**  • Conformed tables: dims/bridges/facts; dedupe + entity resolution.  • AWS: Lake Formation, Glue Data Catalog, Athena, Iceberg on S3.  • OSS: Delta / Iceberg / Hudi; dbt-core for models/tests. | **[4] Ontology & Vocabulary (RDF/RDFS/OWL + SKOS)**  • Model classes/properties; labels & synonyms; map CQs → relations.  • AWS: Store OWL/SKOS in S3; author via notebooks if desired.  • OSS: Protégé; RDF/OWL/SKOS; Jena/RDF4J libs.  **[5] Map Gold → RDF Triples (URIs)**  • R2RML/RML mapping; mint URIs; serialize N-Triples/Turtle.  • AWS: Glue/EMR PySpark jobs write triples to S3.  • OSS: RMLMapper, Ontop, Jena RIOT, Python rdflib.  **[6] Validate & Reason (SHACL + reasoner)**  • SHACL shapes gate data; optional OWL reasoning/consistency.  • AWS: Run pySHACL/Jena on Glue/EMR/Batch; report to S3/CloudWatch.  • OSS: pySHACL, Jena SHACL; HermiT/ELK reasoners.  **[7] Load Knowledge Graph (Graph DB)**  • Bulk load RDF; expose SPARQL endpoint; index/monitor.  • AWS: Amazon Neptune (RDF/SPARQL), S3 Bulk Loader, Workbench, CloudWatch.  • OSS: Jena TDB2 + Fuseki, Blazegraph\*, Neo4j+n10s† (\*unmaintained, †not fully OSS). | **[8] Retrieval indexes (RAG evidence)**  • 8A Keyword/BM25 docs; 8B Vector embeddings for passages.  • AWS: OpenSearch (BM25 & kNN), Bedrock/SageMaker embeddings.  • OSS: OpenSearch self-managed; Qdrant/Milvus/FAISS; sentence-transformers.  **[9] Orchestrate Query → Answer (SPARQL & Federated)**  • Entity link via SKOS; route: SPARQL (facts) | Vector RAG (open-ended) | Federated SPARQL.  • AWS: API Gateway + Lambda/Fargate/EKS; Neptune SPARQL; Step Functions.  • OSS: FastAPI, LangChain/Haystack, Jena/RDFLib clients.  **[10] Present & Observe**  • UI + exports; dashboards; query logs; SPARQL smoke tests (CQs).  • AWS: Amplify/CloudFront/S3, CloudWatch/X-Ray, Athena on logs.  • OSS: React/Vue, Grafana/Prometheus, RDFUnit, pytest. |

Legend: RDF/RDFS/OWL & SKOS → [4]; SHACL → [6]; SPARQL (incl. Federated) → [7],[9]; KG (Neptune) → [7]; RAG Indexes → [8]

Generated 2025-08-31 16:51 UTC