

GraphRAG: Modelos de Lenguaje Razonando con Grafos

Sebastián Ferrada

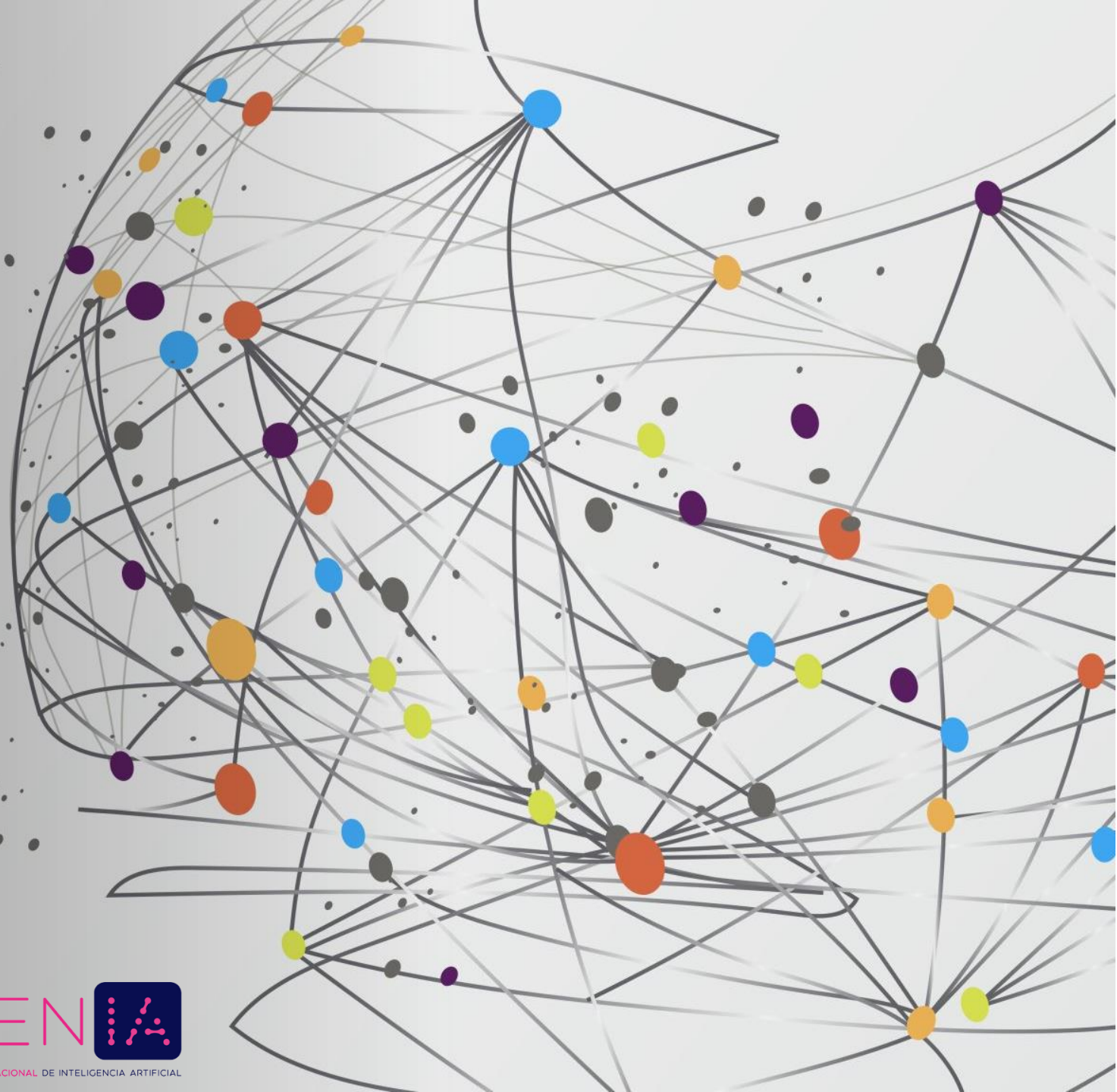
EVIC 2025 – UAndes, Santiago, Chile

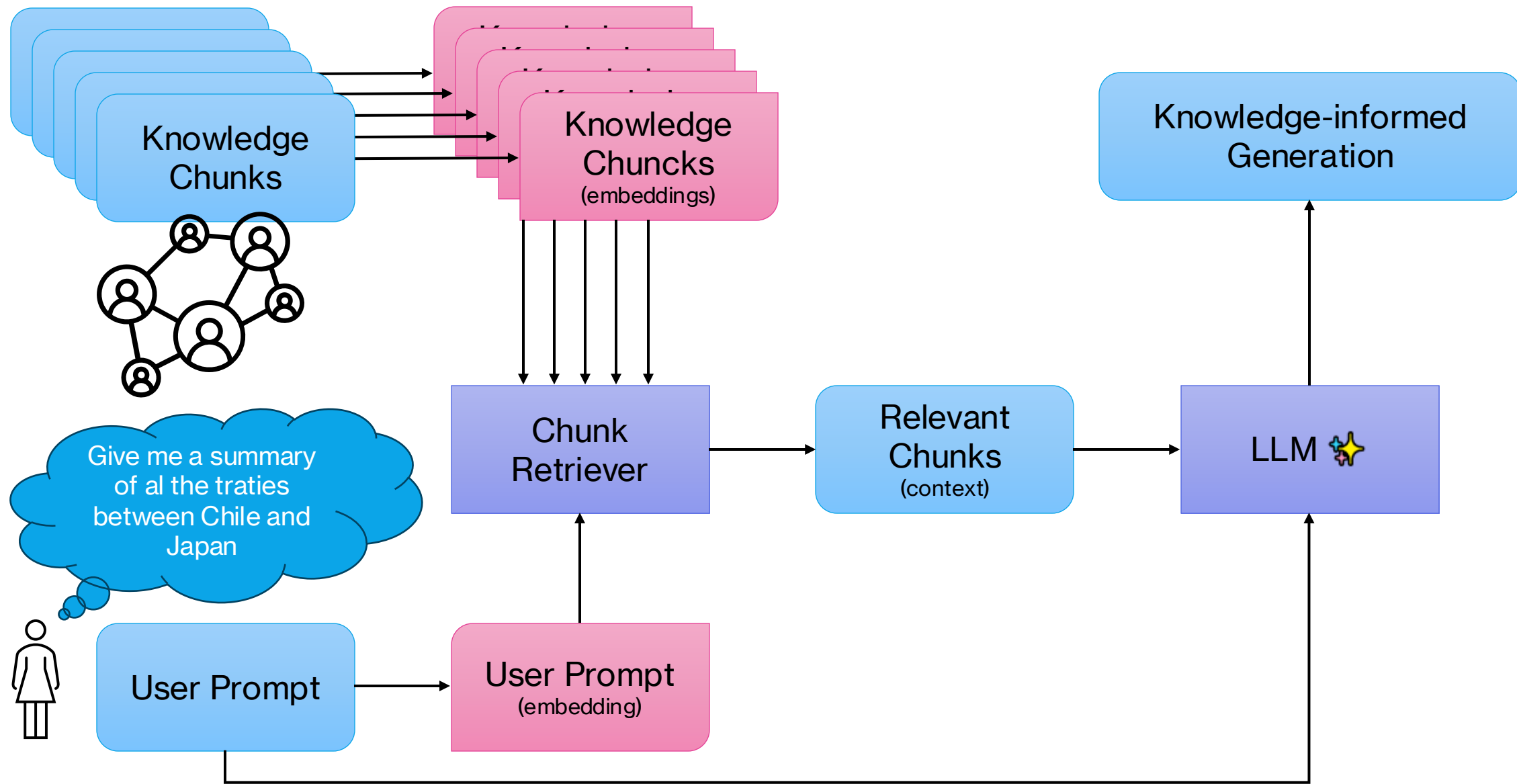


Iniciativa de Datos e
Inteligencia Artificial



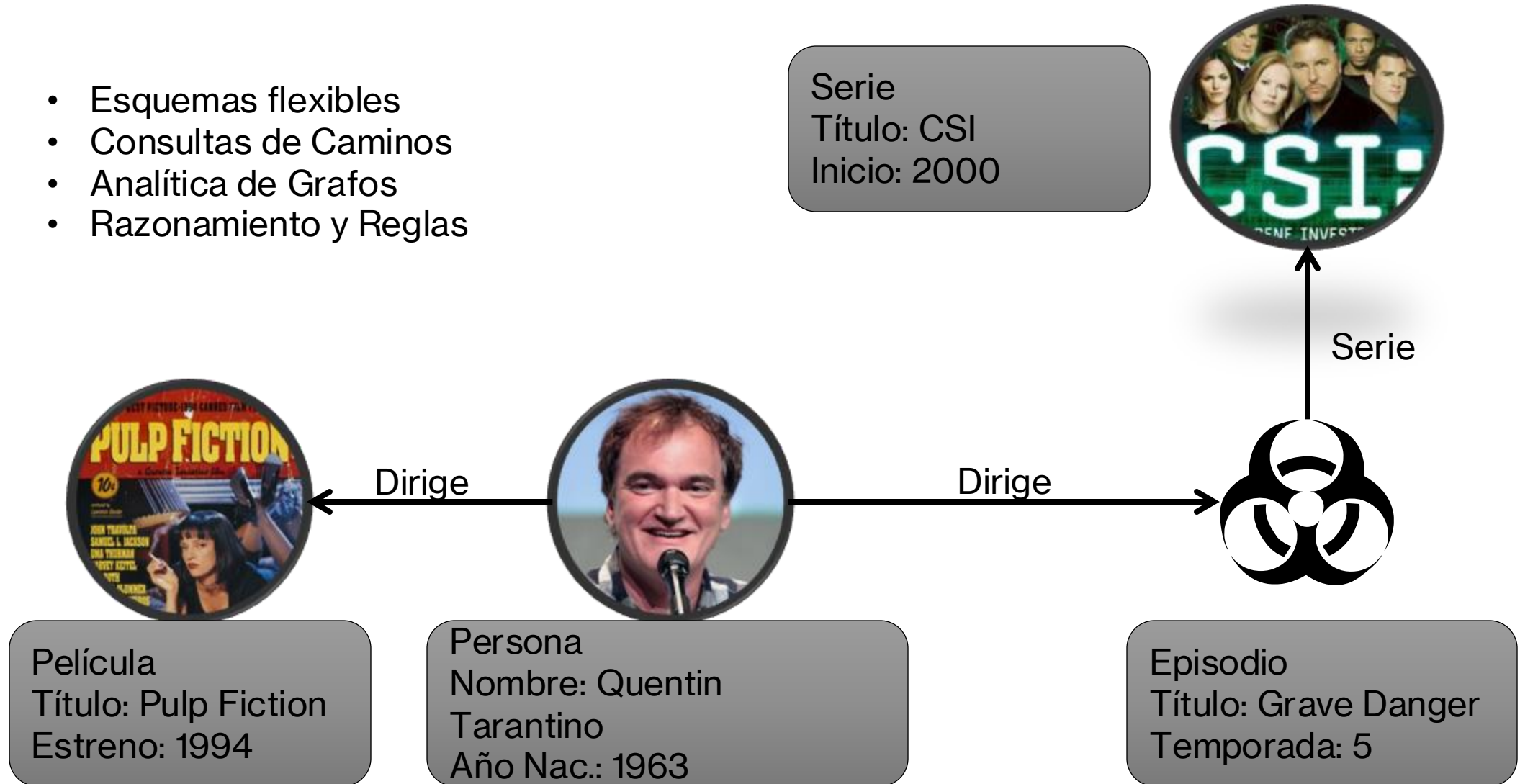
Instituto Milenio
Fundamentos
de los datos





Grafos de Conocimiento

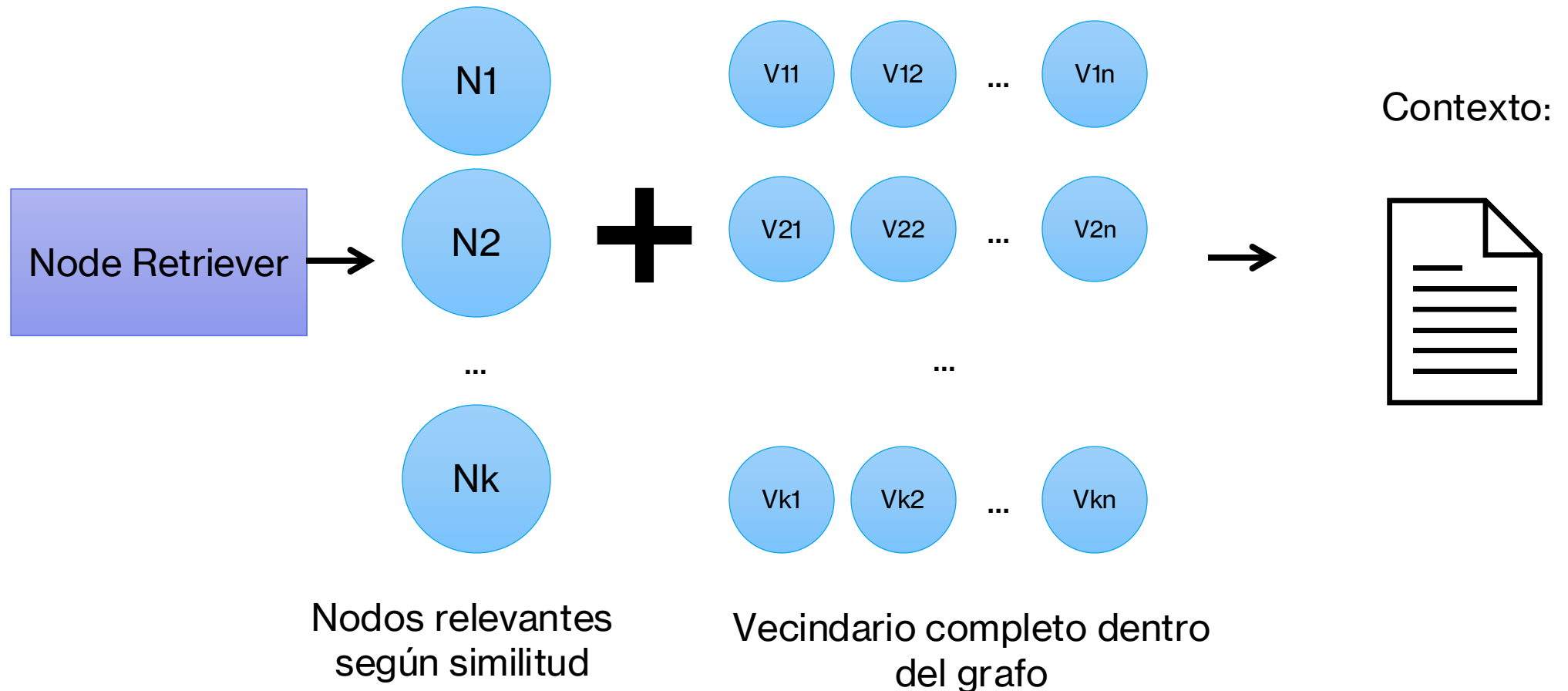
- Esquemas flexibles
- Consultas de Caminos
- Analítica de Grafos
- Razonamiento y Reglas



RAG vs GraphRAG

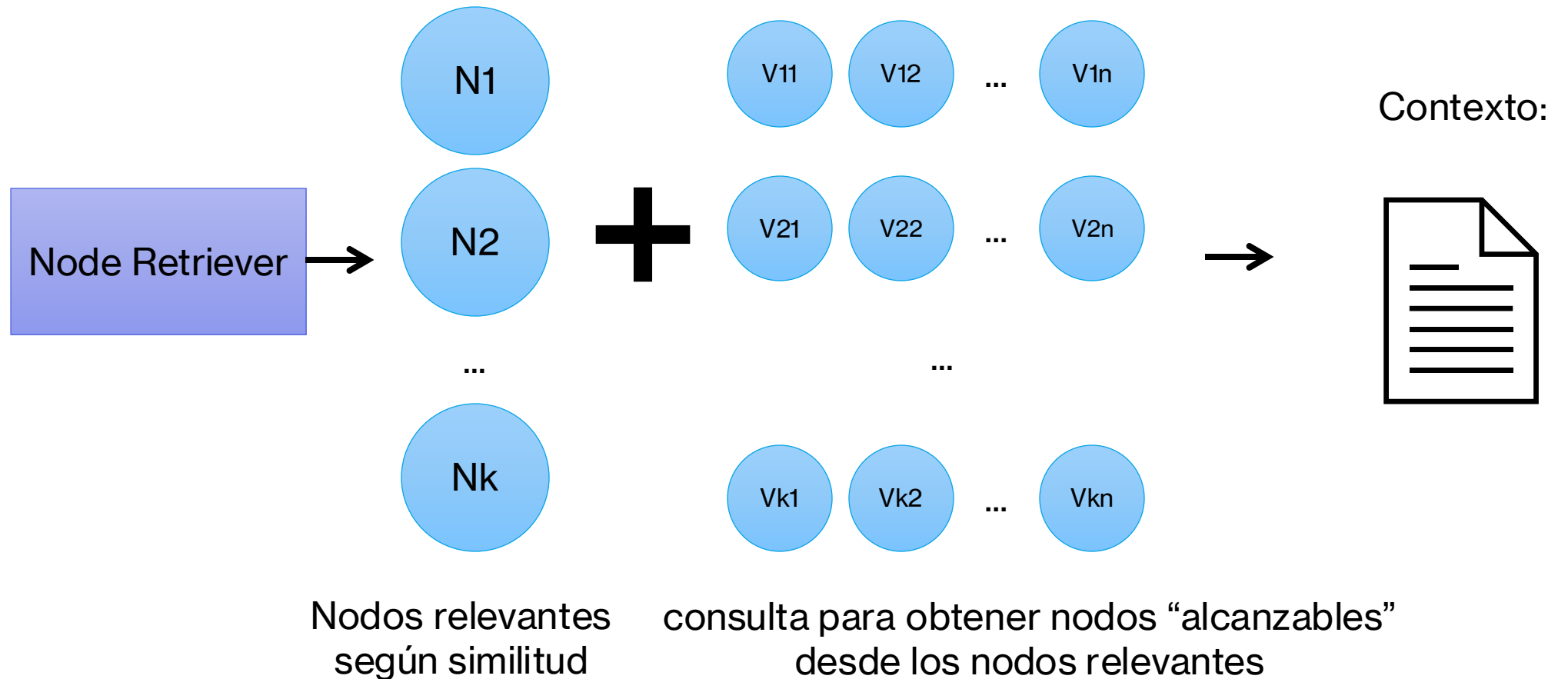
Aspecto	RAG clásico (Vector-based)	GraphRAG
Fuente de información	Fragmentos de texto (chunks) embebidos en un vector store	Entidades y relaciones en un grafo, más embeddings opcionales
Unidad de recuperación	Documentos o párrafos	Nodos, subgrafos, vecindarios, caminos
Estructura del conocimiento	Implícita (chunkeo)	Explícita (propiedades, relaciones, tipologías)
Procesamiento previo	dividir/embeber documentos	modelar grafo, opcionalmente embeber nodos
Recuperación	kNN semántico	búsqueda combinada: vectorial + patrones Cypher + reglas sobre el grafo
Contexto entregado al LLM	Texto concatenado	Texto serializado desde subgrafos relevantes
Ventajas	Simple, plug-and-play	Preciso, explicable, permite razonamiento estructural
Limitaciones	No captura relaciones complejas	Requiere modelado y consultas más elaboradas
Casos ideales	QA superficial sobre documentos	QA profundo, reasoning, data interlinked, relaciones explícitas

Pre-generación de consultas Cypher – Vecindario Completo



Pre-generación de consultas Cypher

– Vecindario relevante al prompt



Pre-generación de consultas Cypher

Ventajas:

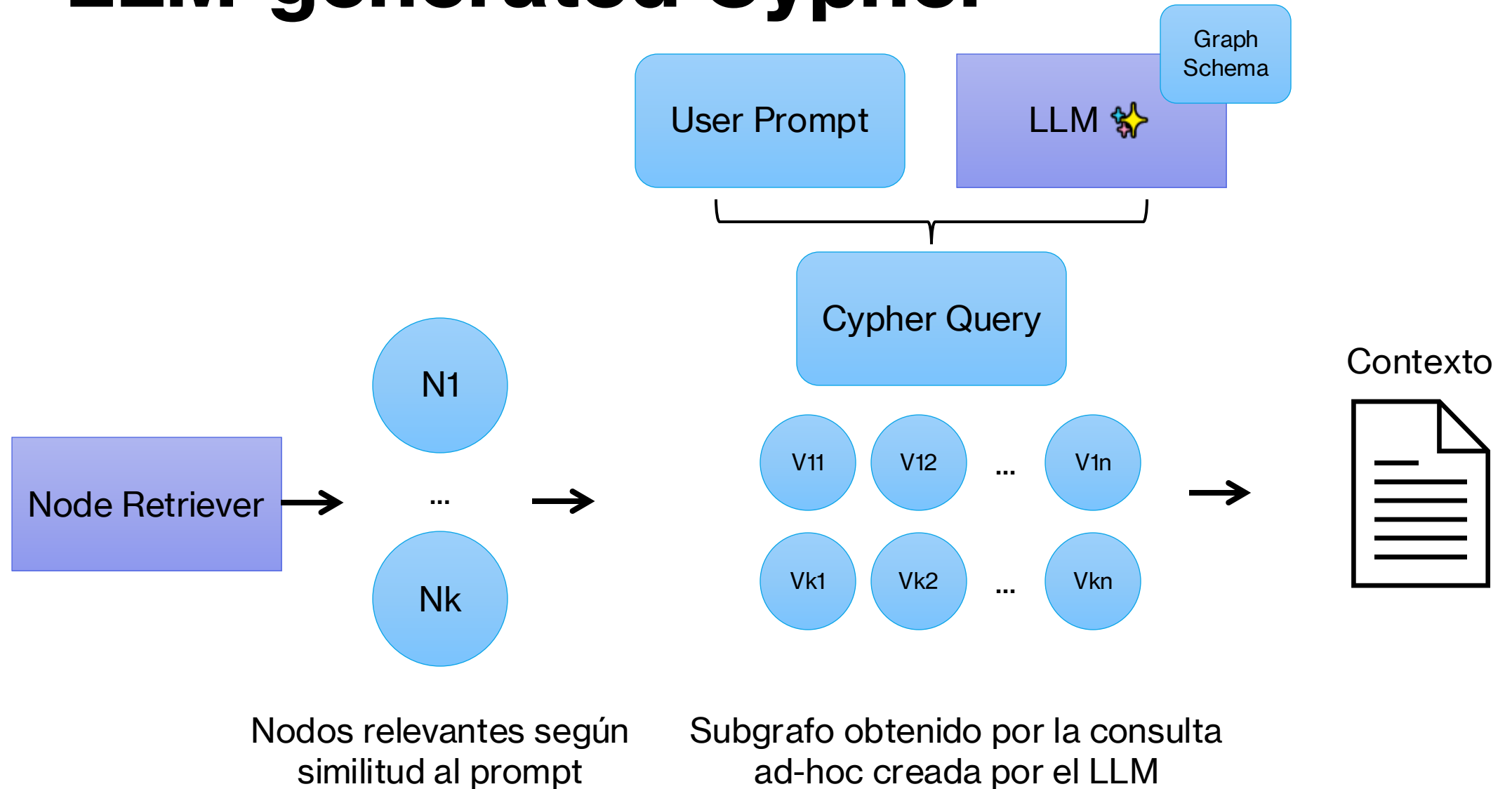
- Fácil de mantener.
- Control total del contenido del contexto.
- Resultados estables y explicables.

Limitaciones:

- No se adapta automáticamente a preguntas complejas no previstas.
- Necesita diseño manual de plantillas Cypher.

Casos ideales: dominios estables, aplicaciones donde quieres consistencia y seguridad (gobierno, empresas, académico).

LLM-generated Cypher



LLM-generated Cypher

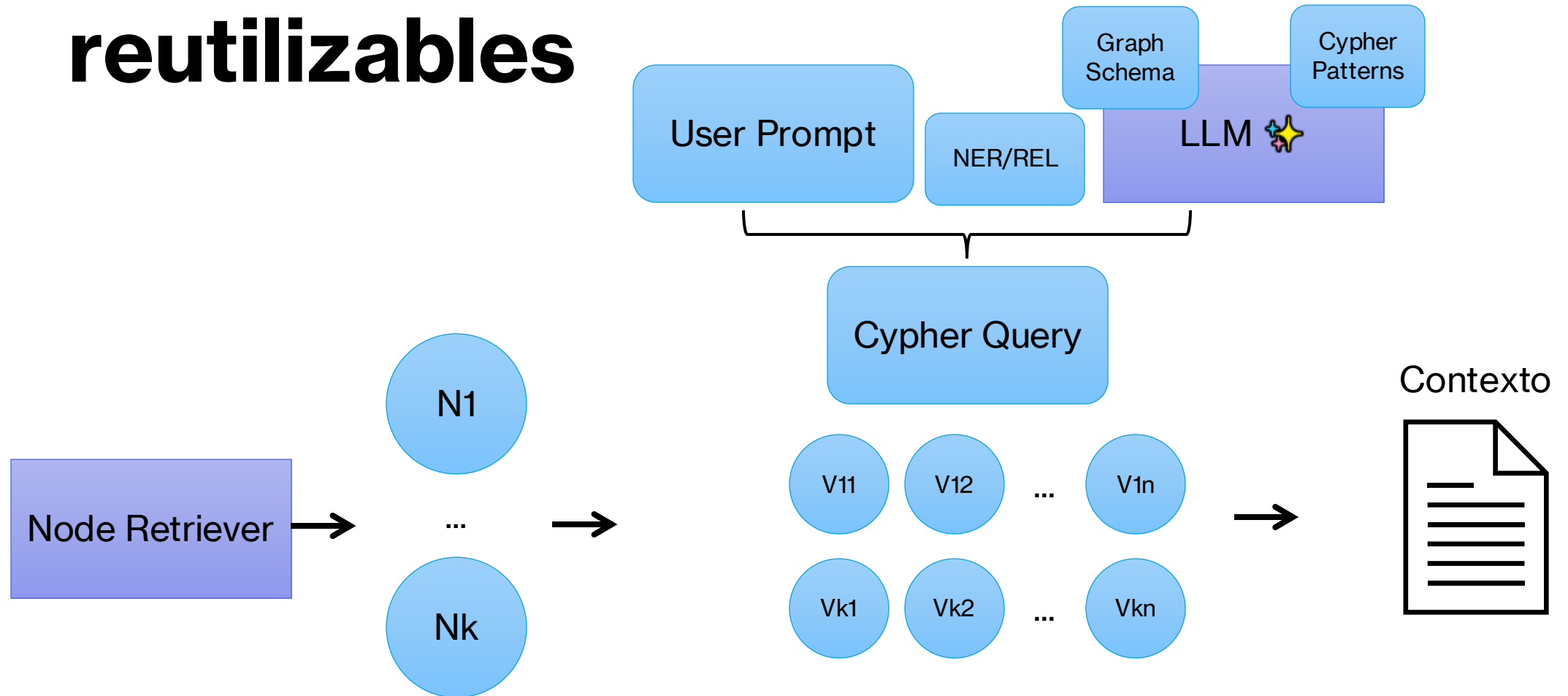
Ventajas:

- Máxima flexibilidad.
- Permite reasoning estructural profundo.
- Se adapta a consultas arbitrarias.

Riesgos:

- Necesita *Schema Guardrails*: validación de Cypher antes de ejecutar.
- El LLM puede generar consultas ineficientes o incorrectas sin supervisión.
- No siempre óptimo en rendimiento.

Método híbrido: patrones Cypher reutilizables



Nodos relevantes según similitud al prompt

Subgrafo obtenido por la consulta ad-hoc creada por el LLM, considerando solo los patrones “permitidos”

Método híbrido: patrones Cypher reutilizables

Ventajas:

- Más seguro que permitir Cypher libre.
- Más flexible que usar solo workflows fijos.
- Fácil de extender.

Limitaciones:

- Requiere mantener un catálogo de patrones.
- Depende de la Calidad del NER/REL (REL puede ser muy malo aun)
- No cubre consultas muy inusuales.

Tutorial



<https://tinyurl.com/GraphRAG-EVIC>