Quick Defrief J1

Cartographie des leviers d'optimisation pour les 12 dernières heures

Milliman × Université Gustave Eiffel

4–5 juin 2026 · 36h · 2 jours

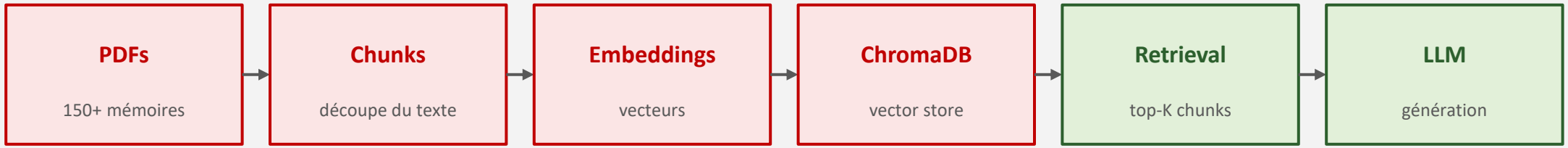
Milliman

Université Gustave Eiffel

Milliman

Le principe à retenir

Toutes les optimisations n'ont pas le même coût



ZONE EMBEDDING

Modifier ici = ré-ingestion nécessaire

ZONE LIBRE

Modifier ici = sans coût d'ingestion

Toucher à l'embedding

Changer le texte des chunks, leur taille, le parser PDF, le modèle d'embedding…

→ Tous les vecteurs doivent être recalculés.
→ Ré-ingestion complète (1h+).

Ne pas toucher à l'embedding

Agir en amont (sur la requête), en aval (rerank, LLM, prompt, post-traitement)…

→ Le vector store ne bouge pas.
→ Itérations rapides, sans coût d'ingestion.

Vue d'ensemble des leviers

Deux colonnes, deux philosophies

✗ TOUCHE L'EMBEDDING (ré-ingestion)

- **Parser PDF** (pypdf, pdfplumber, marker, Docling…)
- **Taille / overlap des chunks** (CHUNK_SIZE, CHUNK_OVERLAP)
- **Stratégie de chunking** (semantic, hiérarchique, par section)
- **Modèle d'embedding** (text-embedding-3-large, BGE-M3, E5)
- **Contextual retrieval** (résumé contextuel avant embedding)
- **Multi-vector retrieval** (chunk + résumé, ou HyDE inverse)

✓ N'Y TOUCHE PAS (itérations rapides)

- **Query rewriting / HyDE / multi-query** (reformuler ou augmenter la requête)
- **Top-K / top-K adaptatif / seuil** (remonter plus ou moins de chunks) fonctionne en general bien avec le reranker
- **BM25 + retrieval hybride** (lexical en complément du dense)
- **Reranking (cross-encoder)** (BGE-reranker, Cohere — local)
- **Prompt engineering** (réécriture, few-shot, structuré)
- **Routing petit / gros modèle** (mini pour simples, full pour complexes)