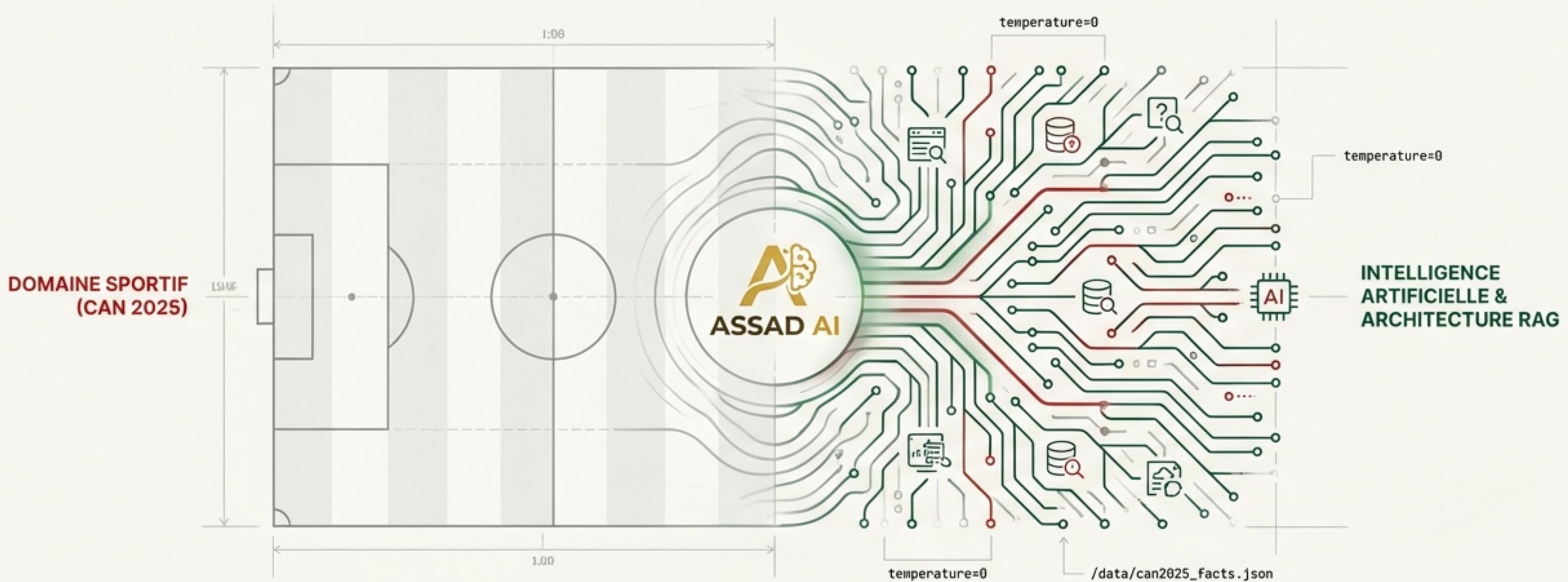


ASSAD AI : L'Architecture d'un Assistant Factuel pour la CAN 2025

Erhensuint une architecture rag:iiete. Intersanminatling au Source Serif Pro

Comment une architecture RAG (Retrieval-Augmented Generation) garantit des réponses précises et fiables pour un événement sportif majeur.



Le Défi : Fournir des Faits, Pas des Fictions

Les LLMs standards, malgré leur puissance, sont inadaptés pour des informations sportives factuelles et évolutives. Leurs limitations posent un risque pour la crédibilité.



Connaissances Statiques

Leur savoir est figé à la date de leur entraînement. Ils ignorent les résultats des matchs à venir.



Hallucinations

Les modèles peuvent inventer des scores, des dates ou des actions de match qui ne se sont jamais produits.

Manque de Traçabilité

Il est impossible de vérifier la source d'une information, rendant chaque réponse potentiellement suspecte.

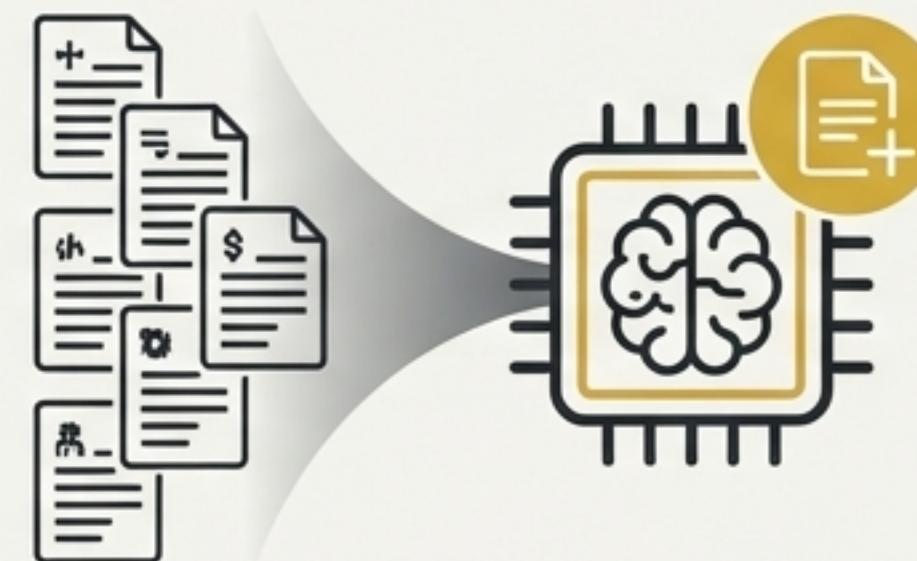
Notre Solution : La Puissance du RAG (Retrieval-Augmented Generation)

Une approche qui ancre la puissance d'un LLM dans une base de connaissances externe et fiable pour garantir la factualité.

1. Récupération (Retrieval)



2. Augmentation (Augmented)



3. Génération (Generation)

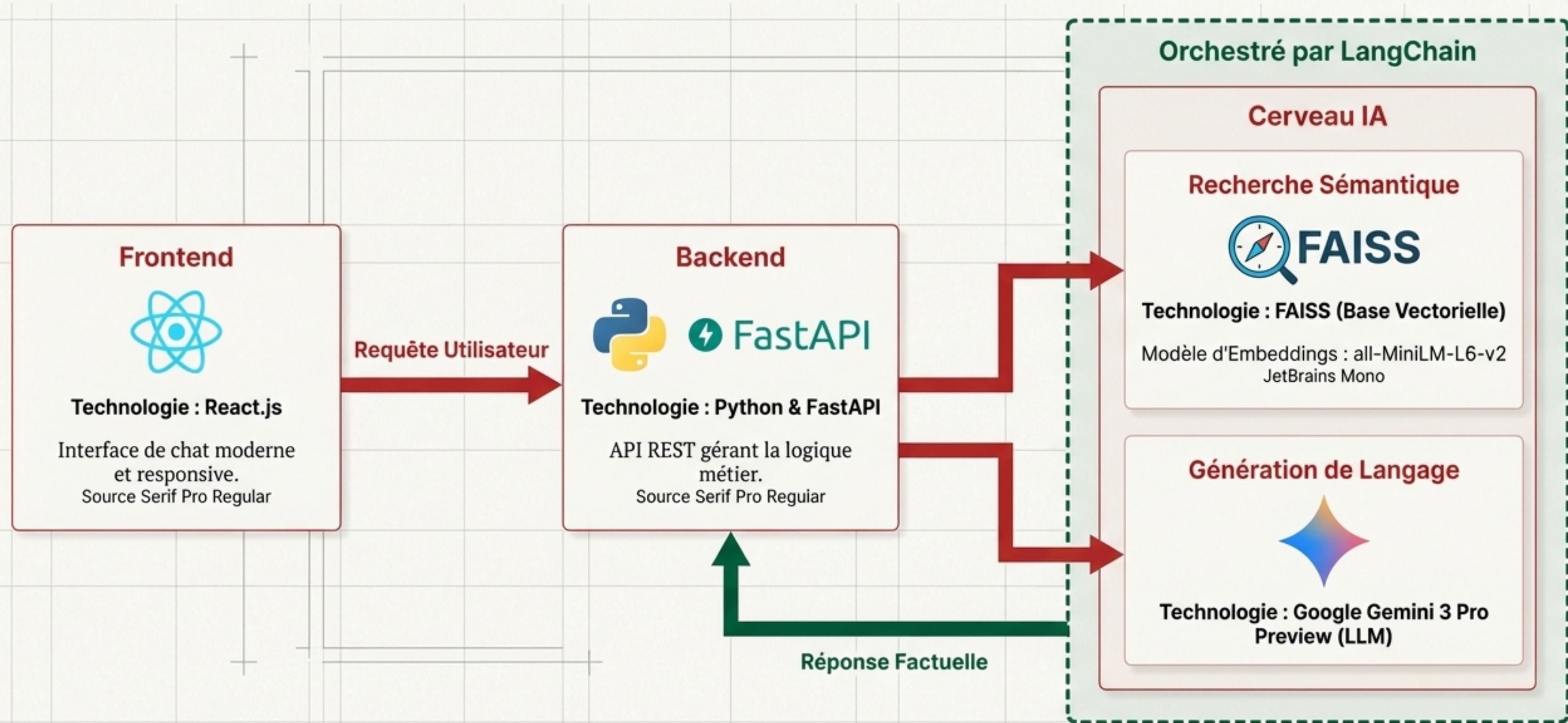
Le LLM génère une réponse naturelle en se basant *uniquement* sur les informations fournies dans le contexte.

Le système recherche et extrait les documents les plus pertinents (fiches de match, données des équipes) de la base de données vectorielle.

Ces documents sourcés sont fournis au LLM comme un contexte obligatoire et exclusif.

Le LLM rédige une réponse naturelle en se basant *uniquement* sur les informations fournies dans le contexte.

Vue d'Ensemble de l'Architecture et de la Pile Technologique



Le Cœur du Système : Un Pipeline en Deux Temps

Phase 1 - Indexation (Offline)

Préparation des Connaissances



Une seule fois, avant le déploiement.



Chargement des données du tournoi (fichiers JSON).



Transformation en documents structurés LangChain.



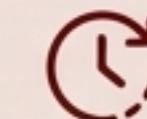
Conversion en vecteurs numériques (embeddings) via `all-MiniLM-L6-v2`



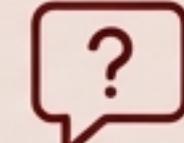
Stockage dans un index binaire `index.faiss` pour une recherche ultra-rapide

Phase 2 - Requête (Runtime)

Génération de la Réponse



À chaque question de l'utilisateur.



La question de l'utilisateur est vectorisée.



Recherche des **k=20** documents les plus pertinents dans l'index FAISS (~5ms)



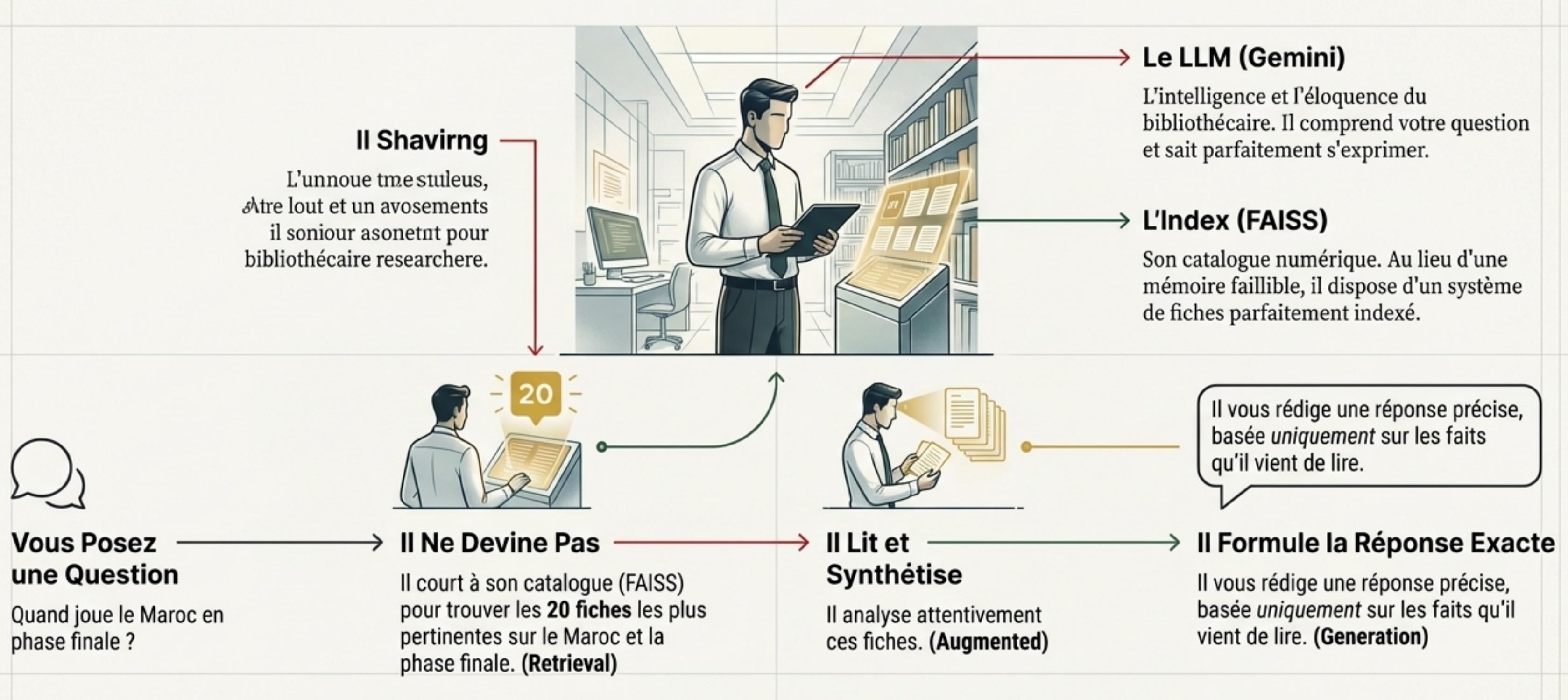
Les 20 documents sont injectés comme contexte dans le prompt.



Le LLM Gemini génère une réponse basée sur ce contexte.

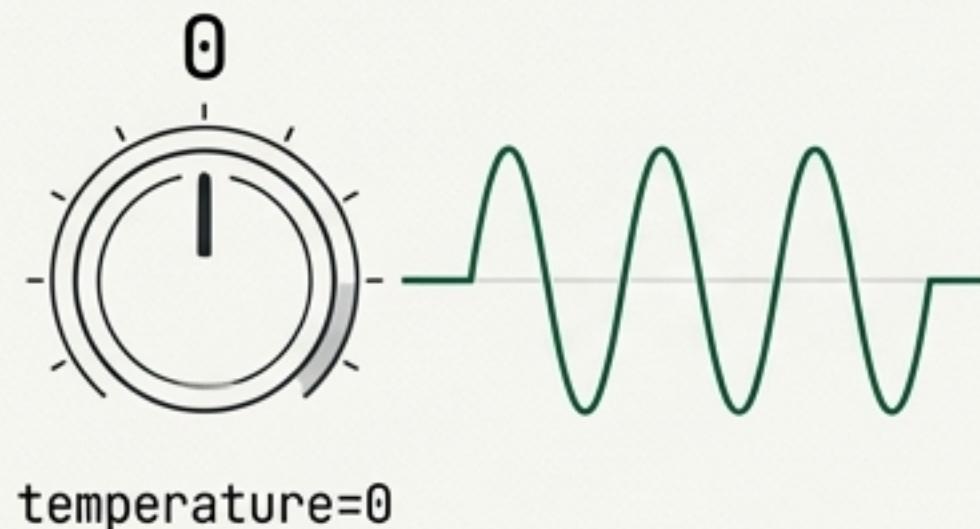
Pour Simplifier : Pensez à un Bibliothécaire Expert

Source Serif Pro Regular, inditation módet neither , un professional researcher



Les Principes Directeurs : Des Choix Techniques au Service de la Précision

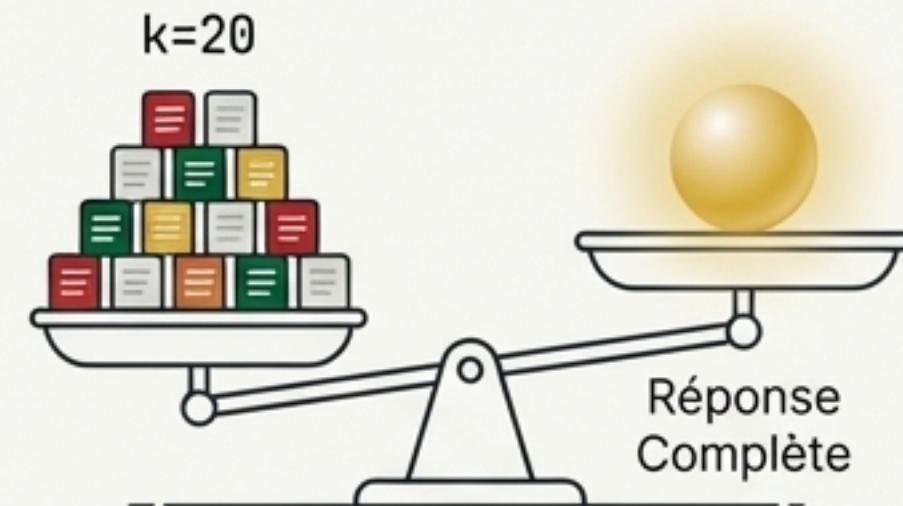
Principe 1 : Déterminisme Absolu



Configuration du LLM avec une `temperature=0`.

Pour les données sportives (scores, dates, noms), la créativité est un défaut. Une même question doit toujours produire la même réponse factuelle. La reproductibilité est essentielle.

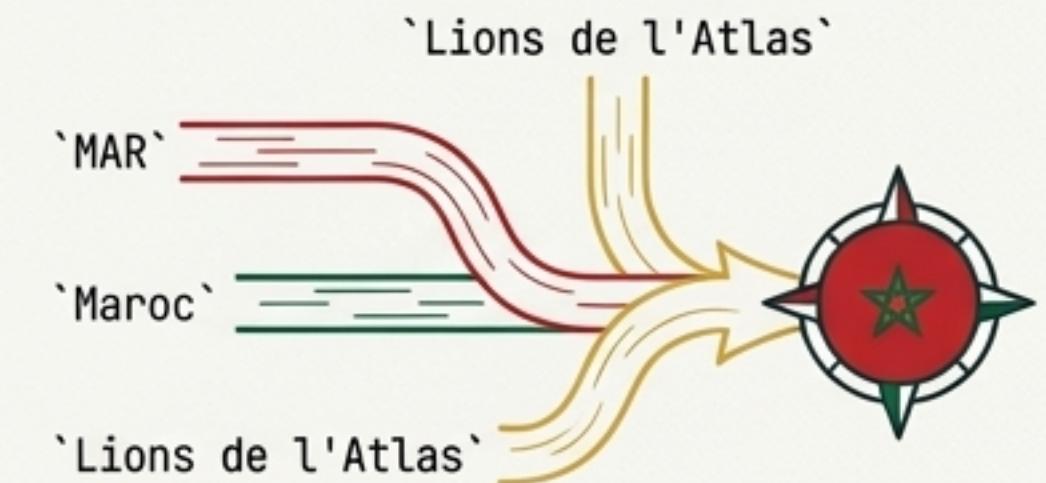
Principe 2 : Contexte Optimal



Récupération systématique des `k=20` documents les plus pertinents.

C'est l'équilibre parfait entre fournir un contexte assez riche pour une réponse complète (gestion des données fragmentées) et maintenir une performance élevée en limitant le nombre de tokens à traiter.

Principe 3 : Reconnaissance Flexible



Système d'alias dans `config.py` (ex: 'MAR', 'Maroc', 'Lions de l'Atlas' sont normalisés).

L'assistant doit comprendre l'intention de l'utilisateur, quelle que soit la terminologie utilisée pour désigner une équipe, garantissant une expérience utilisateur fluide.

Une Performance Mesurée : Rapide, Léger et Efficace

Temps de Réponse Total

~2-3 secondes



De la soumission de la question par l'utilisateur à la réception de la réponse complète.

Génération LLM

~1-2 secondes



Le temps nécessaire à l'API Gemini pour générer la réponse à partir du contexte fourni.

Recherche Vectorielle

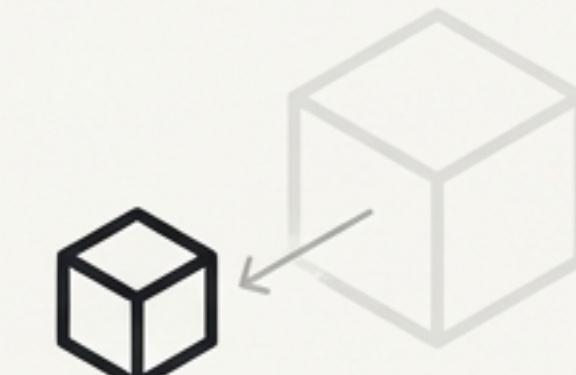
~5 millisecondes



La vitesse à laquelle FAISS identifie les 20 documents pertinents. Le goulot d'étranglement n'est pas la recherche.

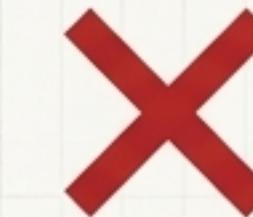
Empreinte sur Disque

~50 MB



La taille de l'index `index.faiss`, démontrant une solution légère et facilement déployable.

Un Périmètre Clair : Ce que l'Assistant Fait (et ne Fait Pas)



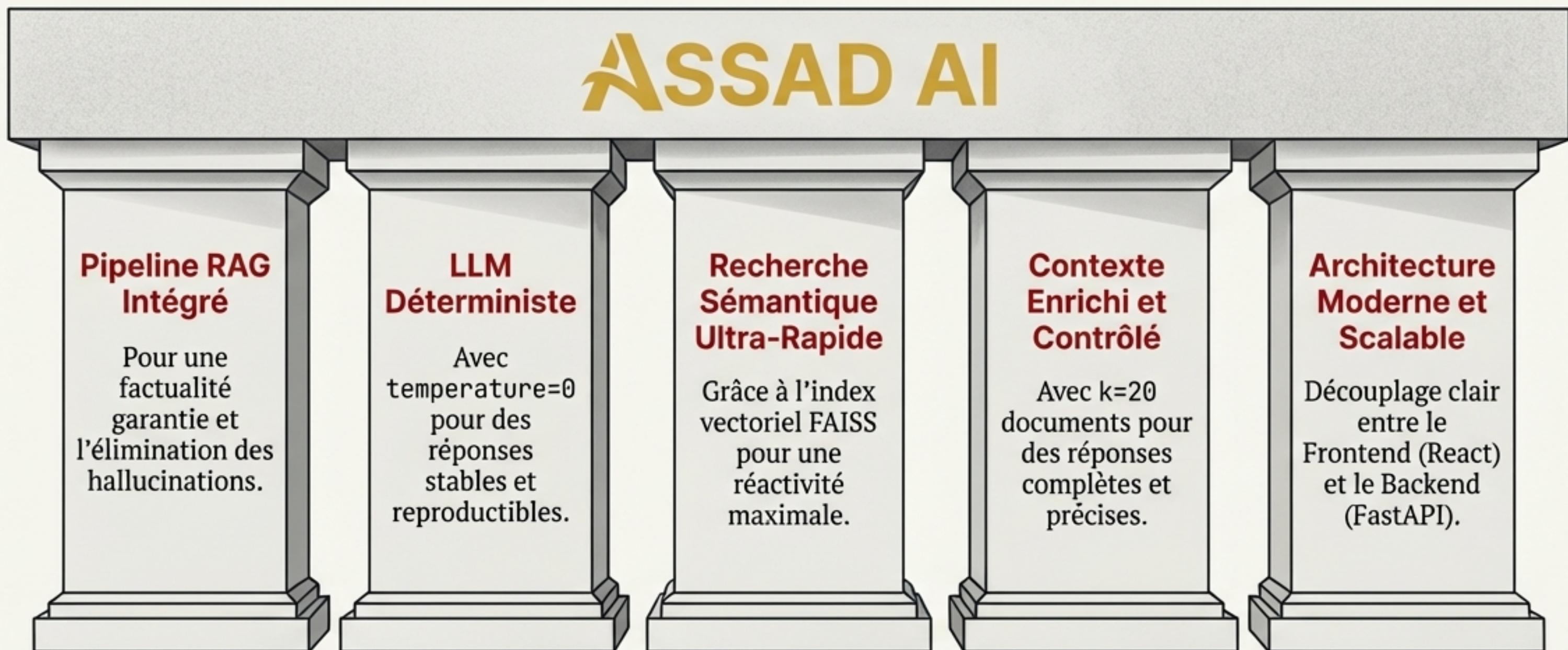
Cas d'Usage Maîtrisés

- ✓ **Questions Factuelles:** Scores, dates, lieux, compositions d'équipes.
- ✓ **Requêtes sur le Calendrier:** Matchs d'une équipe, programme d'une journée, phases du tournoi.
- ✓ **Informations sur les Stades:** Capacités, villes.
- ✓ **Comparaisons Simples:** Qui a marqué le plus de buts entre deux équipes (basé sur les données disponibles).

Limitations Intentionnelles

- ✗ **Pas de Prédictions:** Ne donne aucun pronostic sur les résultats des matchs à venir.
- ✗ **Pas de Spéculations ou d'Opinions:** Ne commente pas la performance des joueurs ou les stratégies des équipes.
- ✗ **Strictement Limité à la CAN 2025:** Ne possède aucune information sur d'autres compétitions ou éditions précédentes.

En Résumé : Les Piliers d'une Architecture Factuelle et Robuste



Cette architecture garantit une logique IA cohérente, performante et digne de confiance, spécifiquement adaptée aux exigences de la CAN 2025.