ADR-003 : Claude via SSH plutôt qu'API Anthropic directe

ADR-003 : Claude via SSH plutôt qu'API Anthropic directe

> *Date : 2026 (premiers tests RAG) — Statut : accepté*

Contexte

L'app a besoin d'appeler Claude (Opus + Haiku) pour : génération RAG, HyDE, decompose-query, contextual retrieval, hierarchy LLM, conflict detection, intent classification, deckbuilding.

Volume estimé : 1000-5000 appels Claude par jour en pic d'usage (notamment pendant les ingestions).

Options :

1. **API Anthropic directe** (`https://api.anthropic.com/v1/messages`)
2. **Claude Code CLI via SSH** vers une VM `oracle` (compte personnel Pro/Team)

Décision

Claude Code CLI via SSH vers une VM dédiée.

Raisons :

- **Quota Pro/Team** : compte personnel Anthropic Pro/Team paie un forfait fixe (~$20/mois) avec quota glissant 5h. Beaucoup plus économique que payer à l'API ($/M tokens) pour du volume moyen-élevé.
- **Outil `Read` natif** : Claude Code a un outil `Read` qui lui permet de lire les PNG des règles directement. Pour la "vision inline" (cf. ADR-004), c'est la fonctionnalité-pivot.
- **No-cost marginal** : un appel de plus = pas de coût supplémentaire (dans la limite quota). Permet d'expérimenter (Contextual Retrieval B = 1 appel par chunk = potentiellement 1000s d'appels par PDF) sans angoisse de facturation.
- **CLI maintenu par Anthropic** : pas de risque que la lib bouge sous mes pieds.

Conséquences

Bonnes

- Coût stable et prévisible
- Vision inline gratuite (lecture PNG ad libitum dans la limite quota)
- Outil `Read` directement accessible — pas besoin d'inliner les images en base64 dans les prompts
- Possibilité d'utiliser des system prompts riches sans facturer chaque token

Mauvaises

- **Quota saturable** : si on dépasse, pause obligatoire (cf. `services/claude-quota.ts` + ADR-007 implicite sur la pause/reprise auto). À l'API ce serait juste $$.
- **Latence baseline ~5s** : tunnel SSH + cold start CLI. Pour les calls courts (HyDE, classify), c'est non-négligeable. D'où le timeout 25s configuré pour absorber cette latence.
- **Sécurité SSH critique** : la VM oracle exécute du code Claude. Verrouillage multi-couche obligatoire (cf. ADR sécurité ssh-oracle).
- **Single point of failure** : si la VM oracle tombe, plus de RAG. Pas de fallback API direct (faisable mais pas implémenté).
- **Pas de batching natif** : chaque appel = un SSH. Pour les pools (Contextual B 10 parallèles), on multiplie les SSH simultanés.

Alternative envisagée

- **API Anthropic directe** : plus rapide (latence ~500ms vs 5s SSH), mais coûte 5-10× plus pour le volume actuel. Non retenu pour des raisons économiques.
- **Hybrid** : API pour les calls courts (HyDE, classify) + SSH pour Opus/Read. Plus complexe, gain marginal. Reporté.

Si on doit migrer un jour

- Implémenter un fallback `services/claude-api.ts` qui wrappe `@anthropic-ai/sdk`
- Mettre les credentials dans une env var `ANTHROPIC_API_KEY`
- Toggle via env (`CLAUDE_BACKEND = 'ssh' | 'api' | 'hybrid'`)
- Garder la vision via une upload image base64 dans le prompt (latence + coût supérieurs)

Effort : 1-2 jours si urgent.

Revision #1
Created 2026-05-10 15:19:52 UTC by thymon
Updated 2026-05-10 15:19:52 UTC by thymon