

Vision inline (Claude lit les PNG au moment Q)

Vision inline (Claude lit les PNG au moment Q)

“ Dernière mise à jour : 2026-05-10

Depuis 2026-04-11, la vision n'est **plus** un pipeline d'ingestion. Claude reçoit les images PNG **directement au moment de la question**, uniquement pour la page la plus pertinente du retrieval, via son outil `Read` côté VM SSH.

Zéro appel LLM vision à l'ingestion. Zéro description intermédiaire textuelle. Zéro cache d'image légendée.

Flux côté `answer.ts`

- Group by** (`livret, page`) : après `retrieveChunks()`, group les chunks non-forum par clé composite (`gameName, pageStart`). Garde le meilleur score par clé.
⚠ **La granularité doit être** (`livret, page`) **et pas seulement** `page` : le retrieval mélange des chunks venant de plusieurs livrets (base + extension + ADV + FAQ), chacun dans son propre dossier PNG (`/app/pdfs/images/<slug>/`). La page 4 de HEAT et la page 4 de Heavy Rain sont deux images distinctes — grouper par page seule renvoie la mauvaise image.
- Sélection** : la meilleure paire (`livret, page`) par score est retenue (`MAX_INJECTED_IMAGES = 1`).
Pourquoi 1 et pas 3 : chaque image coûte ~20-30s de lecture côté Claude (Read tool sur PNG 300 DPI). Passer à 2-3 doublait/triplait la latence pour un gain marginal.
- Résolution du fichier** (`resolvePageImageFile()` dans `pdf-images.ts`) :

- `pdftoppm` produit un padding variable (`page-1.png` / `page-01.png` / `page-001.png` selon le total de pages)
 - Le helper lit le dossier une fois et cache `{pageNumber → filename}`
 - Cache invalidé dans `renderPdfPages` quand un nouveau rendu est lancé
4. **Injection dans le userPrompt** : sous un bloc "IMAGES DES RÈGLES", préfixé par le nom du livret pour que Claude sache quel livret il ouvre :

IMAGES DES RÈGLES :

- Heavy Rain, page 4 : `/projet/boardgame-pdfs/heavy-rain/page-04.png`

5. **Instruction SYSTEM_PROMPT** (règle 9) : Claude ouvre l'image avec son outil `Read` **uniquement** quand la question a une dimension visuelle (icônes, couleurs, symboles, schémas, plateau, tuiles...). Sinon il ignore l'image.

⚠️ `--append-system-prompt` (PAS `--system-prompt`)

`claude-ssh.ts` lance Claude Code CLI avec `--append-system-prompt` et **pas** `--system-prompt`. C'est critique :

- `--system-prompt` **remplace** le system prompt par défaut de Claude Code, qui contient le contrat d'utilisation des outils (`Read`, `Bash`, etc.). Sans ce contrat, Claude garde ses outils techniquement disponibles mais n'en invoque plus aucun proactivement, et finit par halluciner une description à partir du texte seul.
- `--append-system-prompt` ajoute notre `SYSTEM_PROMPT` Oracle **par-dessus** le contrat par défaut. Donc Claude voit les chemins PNG et les ouvre réellement.

C'est un piège facile à reproduire — toujours `--append`.

Volume PNG

- Côté container backend : `/app/pdfs/images/<slug>/page-XX.png`
- Côté VM oracle : monté en read-only sous `/projet/boardgame-pdfs/<slug>/page-XX.png` (chemin contrôlé par `VISION_SSH_IMAGES_PATH`, défaut `/projet/boardgame-pdfs`)
- Le `settings.json` oracle (`/home/oracle/.claude/settings.json`) doit avoir ce chemin dans `permissions.additionalDirectories` ET dans `permissions.allow` pour que Claude puisse `Read`.

Frontend : pas de viewer PNG côté user

Le clic sur une citation `[HEAT, p.4 : "..."]` ouvre toujours **le PDF** (pas le PNG) à la bonne page via `PdfCitationViewer`. Le PNG existe uniquement pour Claude côté backend.

Revision #1

Created 2026-05-10 15:20:16 UTC by thymon

Updated 2026-05-10 15:20:17 UTC by thymon