

# Un Claude peut en cacher un autre

L'écosystème Anthropic en 2026

## Présenté par



**Fabien Canu**

Product Engineer & Expert IA, Koality

Cartographie l'écosystème Anthropic au quotidien sur des projets clients  
TPE/PME.



**Pierre Guirriec**


Ingénieur IA, Polaria

Pratique Claude Code en production. Démo Cowork tirée de son usage réel.

# Notre parcours en 7 temps

 Anthropic, l'acteur et sa cadence

 Comment fonctionne Claude, le socle

 Les interfaces

 Les questions qui reviennent

 Tips concrets pour s'y mettre

 Ce qui est encore bancal

 Démo Cowork en usage réel

# Anthropic, l'acteur

Avant de regarder l'outil, regardons la maison



**30**  
**milliards \$**

de revenus annuels en avril 2026 — multiplié par 10 en 1 an

# Anthropic vient de dépasser OpenAI en revenus

**30** Md\$

Anthropic

ARR avril 2026, valorisation 380 Md\$

**25** Md\$

OpenAI

ARR Q1 2026, valorisation 852 Md\$

**200** M\$

Perplexity

ARR février 2026, valorisation 20 Md\$

## Trois indicateurs qui parlent

**1 000+**

grandes entreprises clientes

Chacune paye plus d'1 M\$/an. Cursor,  
Notion, Stripe, BCG...

**Bourse 2026**

entrée prévue à l'automne

Goldman, JPMorgan, Morgan Stanley pilotent  
l'IPO

**10 GW**

compute engagés pour 2027

Équivalent à 7 réacteurs de Cattenom, ou à la  
consommation électrique de la Belgique

# Et tout récent : 220 000 GPU loués par SpaceX

Annoncé le 6 mai 2026. Clin d'œil sur les ordres de grandeur.

**SpaceX loue Colossus 1 à Anthropic.** 220 000 GPU Nvidia, à Memphis. Le data center qu'utilisait xAI pour entraîner Grok.

**Et pourtant : seulement 3 % de leur capacité engagée.** Anthropic a déjà 10 GW au total (Amazon 5 GW, Google 3,5 GW, Azure 1 GW, plus Fluidstack). 0,3 GW de SpaceX, ce sont quelques pourcents en plus.

**Trois mois plus tôt,** Musk traitait publiquement Anthropic d'« evil ». En mai, il leur loue son data center. Le business, ça fait drôle.

# Deux géants soutiennent Anthropic

## Amazon

13 Md\$ déjà investis (jusqu'à 33 Md\$ avec les jalons)

Cloud secondaire avec les puces maison Trainium

Anthropic s'engage en retour à 100 Md\$ d'AWS sur 10 ans

## Google

13 Md\$ déjà investis (jusqu'à 40 Md\$ avec les jalons)

Cloud principal, ses puces TPU entraînent Claude Opus, Sonnet, Haiku

Anthropic s'engage en retour à 200 Md\$ de Google Cloud sur 5 ans



*En 10 ans, Anthropic dépensera plus en serveurs que la mission lunaire n'a coûté.*

— Programme Apollo : ~250 Md\$ en dollars actualisés. Engagements cloud Anthropic sur 10 ans : 300 Md\$. Différence majeure : Apollo, c'était de la dépense pure ; Anthropic, c'est un investissement censé rapporter.

# Anthropic Labs, l'usine à produits



# Claude Code a prouvé. Cowork est le pari suivant.

Même méthode, marché 33 fois plus grand.

## Claude Code

Les développeurs, environ 30 millions dans le monde

1 milliard de dollars de revenus en 6 mois, record absolu pour un produit dev

Produit installé qui rapporte, marché conquis

## Cowork

Tous ceux qui bossent dans un bureau (admin, marketing, finance, RH, juridique...), environ 1 milliard de personnes

33 fois plus gros que le marché des développeurs

Tout juste sorti de version test, le pari démarre maintenant

## Quatre signaux que le pari est sérieux

**Mike Krieger sur le coup.** Le mec qui a cofondé Instagram dirige Cowork depuis janvier 2026. Anthropic met ses **poids lourds sur ce produit**, pas un junior.

**Tout pour les grandes boîtes en mai 2026.** Anthropic a sorti d'un coup ce qu'il faut pour qu'une grande entreprise déploie Cowork sans risque : **qui peut faire quoi, suivi des actions, plafonds de dépenses**. Avant impossible, maintenant oui.

**Trois grosses banques pour pousser le truc.** Goldman Sachs, Blackstone et Hellman & Friedman. Leur boulot : **convaincre les patrons** des grandes entreprises d'installer Claude.

**Microsoft les copie.** Microsoft a sorti son propre Cowork en mars 2026, avec **Anthropic comme partenaire technique**. Quand ton concurrent principal te copie, c'est que tu as trouvé un truc qui marche.

# Microsoft glisse vers Anthropic

Côté Anthropic	Côté Microsoft (concurrent direct)
Claude Chat	Microsoft Copilot Chat — Claude inclus depuis mars 2026
Claude Code	GitHub Copilot — Claude dispo depuis octobre 2024, Opus 4.7 GA le 16 avril 2026
Claude Cowork	Microsoft Copilot Cowork — propulsé par Anthropic depuis mars 2026

*Fin 2023, Microsoft était quasi-exclusivement OpenAI. L'exclusivité formelle a pris fin le 27 avril 2026. Aujourd'hui, Anthropic est intégré dans tous leurs Copilots.*

# Claude, c'est quoi en fait ?

Quand on dit « Claude », on parle parfois du modèle, parfois du produit.  
Selon le contexte.

## Au sens strict

Un **modèle d'IA, un LLM** (Large Language Model)

**Trois variantes** : Claude Opus 4.7, Sonnet 4.6, Haiku 4.5

Un **réseau de neurones** qui prend du texte en entrée et produit du texte en sortie

## Au sens large (langage courant)

**Le produit complet** : Claude.ai, Claude Desktop, l'API...

Le modèle plus **tout ce qu'on a empilé autour** pour le rendre utile (contexte, mémoire, outils)

# Comment fonctionne Claude, le socle

Une métaphore qui éclaire toute la suite.



# Une interaction complète, étape par étape

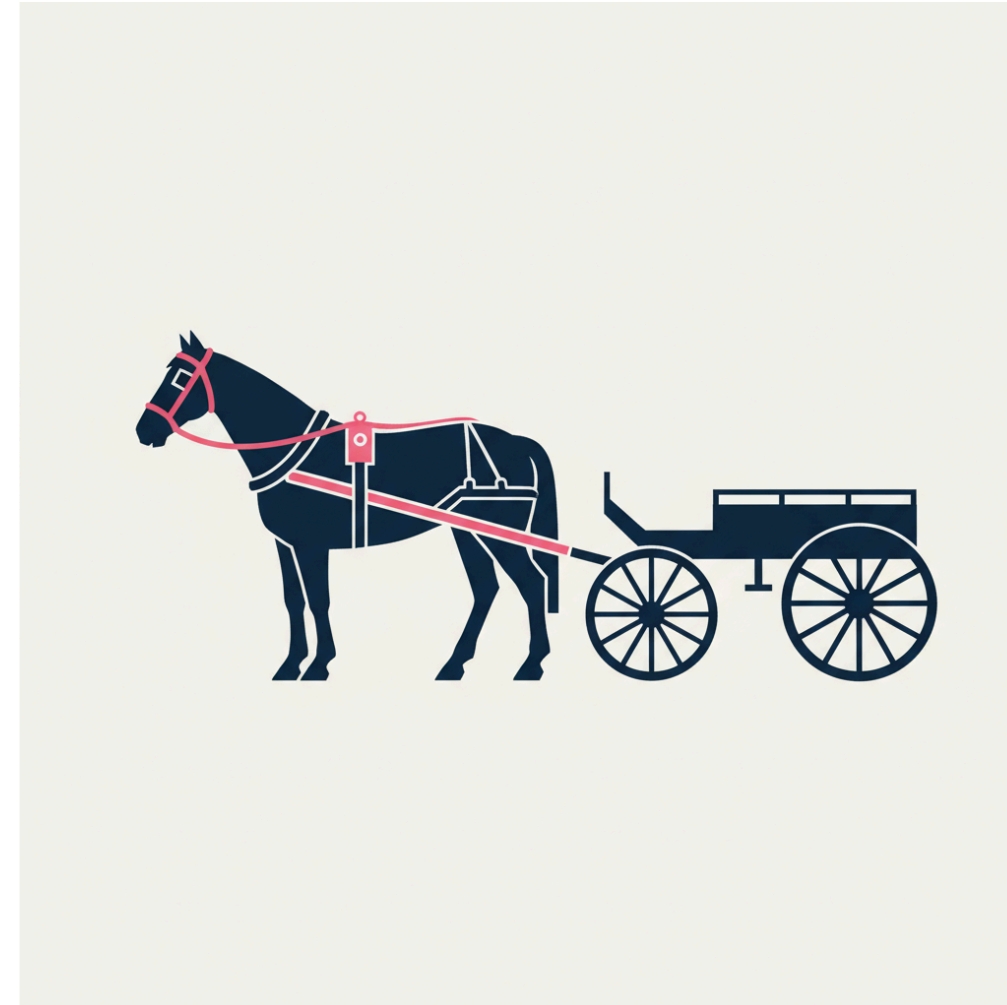


# Le harnais

*Le cheval tire.* (le LLM)

*Le harnais dirige.*  
(le guide)

*La charrette avance.*  
(l'objectif)



# Penser harnais, c'est penser capacités.

Selon où il s'exécute, ses capacités changent.

Sur l'ordinateur (Desktop, Code) : accès à tout.

Dans le navigateur (Claude.ai) : limité.

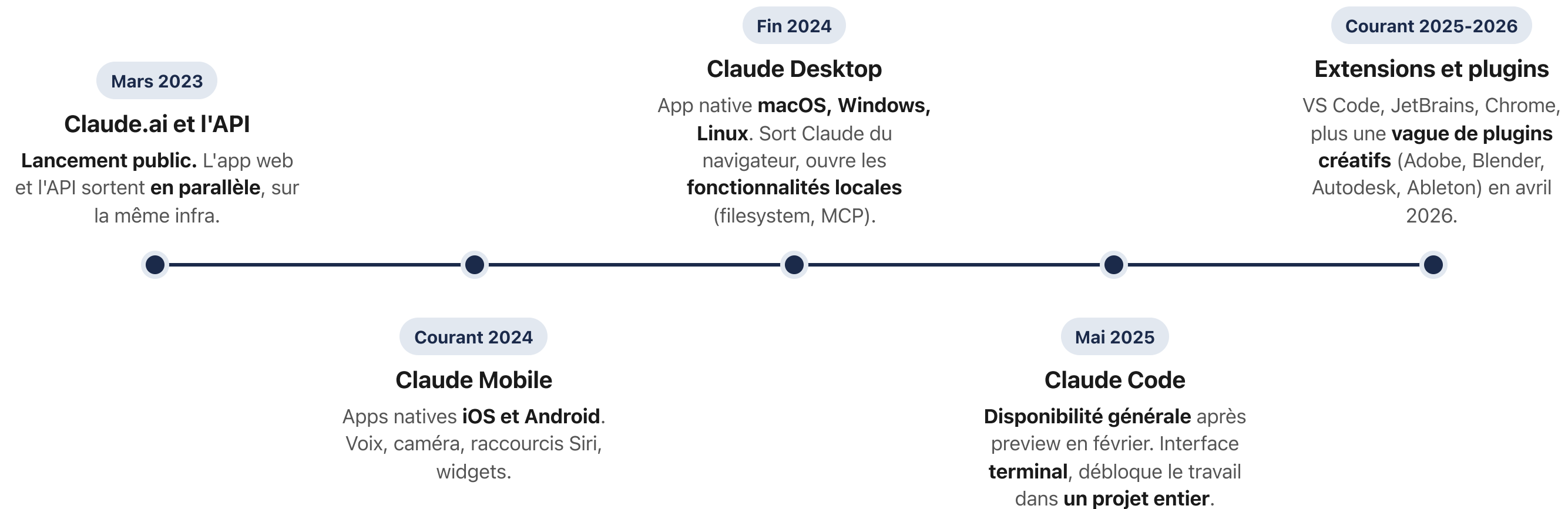
Sur le téléphone (Mobile) : autres limites.

# L'écosystème Claude, interface par interface

Tour des six interfaces, des fonctionnalités qu'elles proposent, et choix d'interface selon l'usage



# Les six interfaces, par ordre d'apparition



# L'écosystème en treize fonctionnalités

Tour rapide des concepts qui circulent autour de Claude. **Chacun sera examiné dans son interface naturelle** ensuite.

Chat

Cowork

Dispatch

Claude Code

Claude Design

Projects

Memory

Skills

Connectors MCP

Plugins

Computer Use

Artifacts

Agent SDK

---

## Agent SDK

Le kit de développement pour construire ses propres agents Claude au-dessus de l'API

# Claude.ai

Un onglet, un chat, le modèle. Le harnais le plus simple.



## Sandbox du navigateur

**Le navigateur décide de tout.** Pas de disque, pas de processus locaux.



## Le harnais le plus léger

**Un tour de chat à la fois.** Pas de chaînage, pas de sub-agents.



## Le point d'entrée

**L'expérience canonique** pour découvrir Claude et l'usage quotidien.

Chat

Co-work

Dispatch

Claude Code *Pro+*

Claude Design *Preview*

Projects

Memory

Skills

Connectors MCP *remote uniquement*

Plugins

Computer-Use

Artifacts

Agent SDK

# Claude Mobile

Claude dans la poche. Le harnais mobilité.



## Sandbox de l'OS mobile

Pas d'accès au filesystem du PC, pas d'exécution de code locale.



## Capacités natives du téléphone

Voix, caméra, raccourcis Siri, widgets. Memory partagée avec Claude.ai.



## Pour la mobilité

Capter en route, dicter, poser une photo. Aussi télécommande pour Dispatch.

Chat

Co-work

Dispatch *côté commande*

Claude Code *via Dispatch*

Claude-Design

Projects

Memory

Skills

Connectors MCP *remote uniquement*

Plugins

Computer Use *via Dispatch*

Artifacts

Agent-SDK

# Claude Desktop

Claude installé sur le poste. Le harnais qui voit vos fichiers.



## App native macOS et Windows

Accès au filesystem local, lance des outils, embarque une **VM Linux pour Cowork**.



## Le harnais GUI le plus complet

Cowork, Dispatch, Connectors MCP locaux, Plugins. Le pivot hors navigateur.



## Pour le knowledge work assisté

Sortir du navigateur, garder ses fichiers à portée, piloter d'autres apps.

Chat

Cowork *Pro+*

Dispatch *côté exécution*

Claude Code *intégré*

Claude Design

Projects

Memory

Skills

Connectors MCP

Plugins

Computer Use *Pro/Max*

Artifacts

Agent-SDK

# >\_ Claude Code

Le harnais agent complet. Tourne dans un projet.



## Hôte = terminal natif

Aussi disponible dans **Desktop, VS Code, JetBrains**. Accès filesystem complet.



## Tout l'arsenal agentique

Boucle profonde, sub-agents, hooks, Plugins, MCP, Computer Use, Agent SDK.



## Pour développer et automatiser

Le harnais le plus puissant côté outils. CLAUDE.md comme mémoire versionnée.

Chat *TUI*

Cowork

Dispatch *côté exécution*

Claude Code *natif*

Claude Design *handoff entrant*

Projects *via dossier*

Memory *CLAUDE.md*

Skills *filesystem*

Connectors MCP

Plugins *natif*

Computer Use

Artifacts

Agent SDK

# Extensions et plugins

D'autres façons d'utiliser Claude. Pas un harnais autonome, une greffe sur un hôte.



## L'hôte fixe les règles

Ce que Claude peut faire dépend de ce que l'app hôte autorise.



## Deux logiques opposées

Soit **Claude s'invite dans l'hôte** (VS Code, JetBrains, Chrome, Slack), soit **Claude pilote l'hôte de l'extérieur** (Adobe, Blender, Autodesk via Plugins MCP).



## Pour rester dans son flux

**Pas besoin de quitter son outil métier.** Ou inversement : Claude au centre, les apps autour.

Chat *dans l'hôte*

Cowork

Dispatch

Claude Code *VS Code, JetBrains*

Claude-Design

Projects *via compte*

Memory *via compte*

Skills *selon hôte*

Connectors MCP *limités*

Plugins

Computer-Use

Artifacts

Agent-SDK

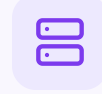
# <> API Anthropic

Le harnais que vous codez. Brique nue.



## Aucune UI fournie

**Tout à bâtir** : boucle, contexte, mémoire, outils, permissions.



## Le sous-sol de l'écosystème

**Toutes les autres interfaces sont bâties au-dessus.** Memory tool natif, prompt cache, batches.



## Pour développeurs et plateformes

**Aussi via AWS Bedrock, Google Vertex, Azure AI Foundry** pour rester sur son cloud.

Chat *à coder*

Cowork

Dispatch

Claude Code *via Agent SDK*

Claude Design

Projects *à coder*

Memory *memory tool*

Skills *workspace*

Connectors MCP

Plugins

Computer Use

Artifacts

Agent SDK *natif*

# Ce que tout le monde se demande

Quatre questions qui reviennent dans toutes les salles.



# Skills vs MCP

Deux mécaniques du socle, exploitées par plusieurs harnais.

## Skills

Branche sur la brique CONTEXTE du socle.

Un dossier de fichiers (instructions + scripts) que Claude lit quand il en a besoin.

Vit dans le filesystem du harnais (Code, Desktop, API workspace).

Statique : Claude charge, applique, continue.

Exemple : skill « PDF Koality » qui explique comment générer un PDF.

## MCP

Branche sur la brique OUTILS du socle.

Un protocole standard pour brancher des serveurs sur Claude.

Deux modes : stdio (sous-processus local) ou HTTP (URL distante).

Dynamique : Claude appelle, reçoit, recommence.

Exemple : MCP Notion qui expose lire/créer/modifier une page.

# Dispatch

L'intention voyage d'un harnais à un autre.

L'intention envoyée depuis ailleurs : tu dictes une tâche depuis Claude Mobile, ton Mac la reçoit et l'exécute.

Le Mac route l'intention vers Cowork ou Claude Code selon la nature de la tâche (knowledge work ou code).

Tourne dans Claude Desktop, qui doit être ouvert sur ta machine. Plans Pro et Max, lancé mars 2026.

# Scheduler vs Routines

Deux mécaniques pour faire tourner Claude périodiquement. Mais pas au même endroit, et pas pour les mêmes usages.

## Scheduler

**Tourne sur ton Mac**, dans Cowork (Desktop).

**Ton Mac doit être allumé et Cowork ouvert.**

Triggers : **hourly, daily, weekly, weekdays, ou manuel.**

Accès à tes **fichiers locaux, connectors, plugins, skills** Cowork.

Exemple : \*« Tous les lundis 9h, compte-rendu Slack de la semaine. »\*

## Routines

**Tourne dans le cloud Anthropic**, dans Claude Code.

**Ton laptop peut être fermé.** Ça tourne quand même.

Triggers riches : **cron, webhook GitHub (push, PR, issue), appel API.**

**Accès au repo, MCP, shell.** Un agent complet, pas juste un cron-script.

Exemple : \*« À chaque PR ouverte, lance la revue automatique. »\*

# Computer Use : la promesse

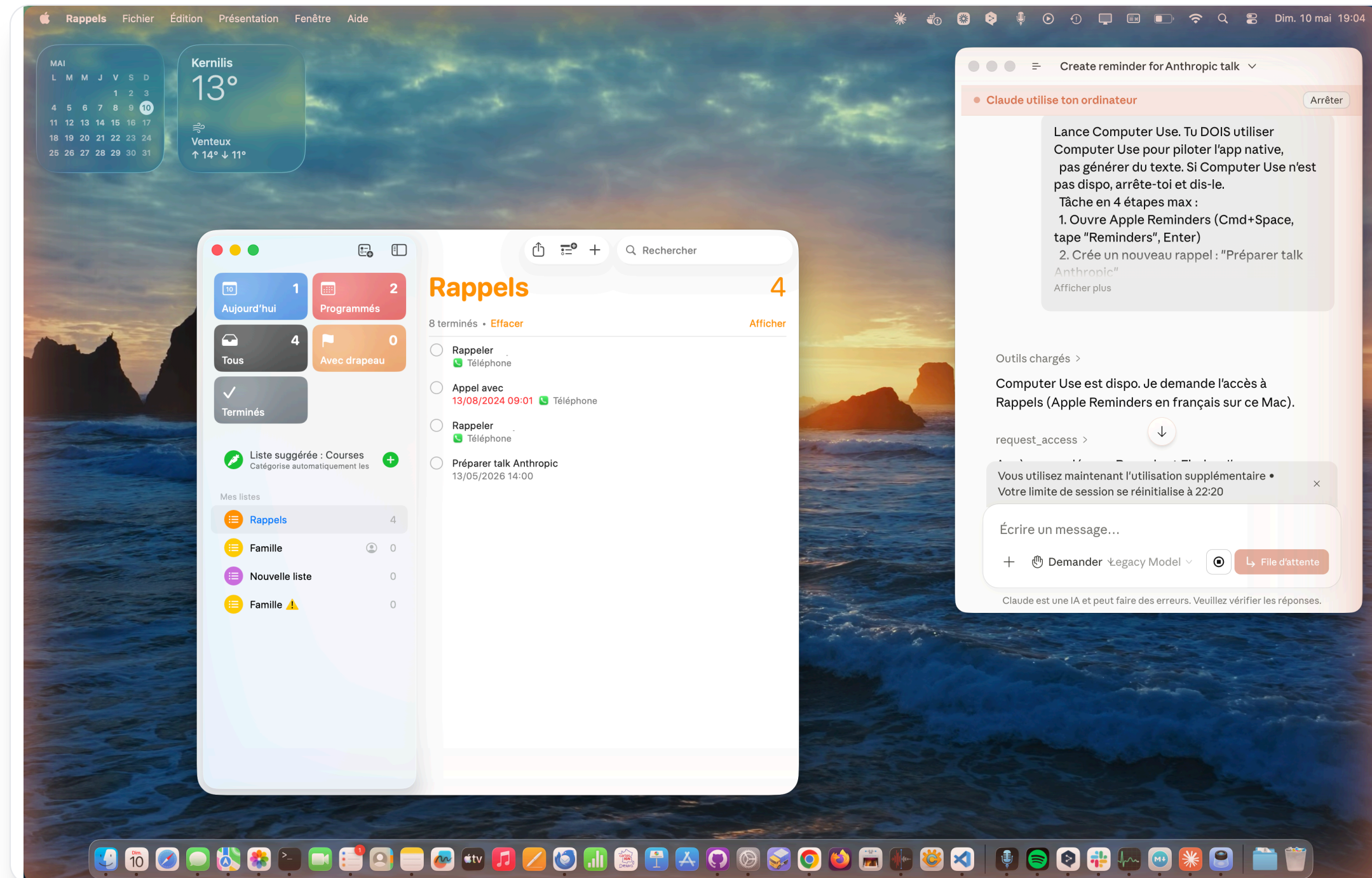
Le harnais qui pilote votre écran.

Tu décris l'objectif. Claude pilote ton ordi à ta place : il bouge la souris, tape au clavier, lit l'écran.

Pas d'API à coder, pas de scraping. N'importe quelle app qui marche pour toi marche pour Claude.

Réservation de billets, formulaire administratif, comparatif de produits, automatisation d'app sans API : tout devient automatisable.

# Sur app native, ça marche. Lentement, mais ça marche.



Le rappel créé, comme demandé

# Sur le web : DOM dynamique, et ça casse.

Find train Brest to Paris Thursday morning

browser\_batch >

Je règle l'heure d'abord (tant que le calendrier est ouvert), puis Confirmer.

A utilisé Claude dans Chrome (3 actions) >

Dropdown ouvert avec 08h visible. Je clique 08h puis Confirmer, et je passe aux villes derrière.

A utilisé Claude dans Chrome, outil utilisé >

Heure 08h posée. Un popup "Qui voyage ?" s'est ouvert : je le ferme et je corrige les villes utilisant les refs DOM cette fois (plus fiable).

browser\_batch >

Je clique directement sur le champ Départ pour basculer le focus dessus.

A utilisé Claude dans Chrome (2 actions) >

"Brest (Bretagne) Ville" est en haut de la liste, highlighted. J'appuie Enter pour valider la suggestion, c'est plus fiable qu'un clic.

browser\_batch >

Brest posé. Le calendrier s'est ouvert avec 14 mai/08h corrects. Je Confirme et je passe à Arrivée.

A utilisé Claude dans Chrome (2 actions) >

You've hit your limit · resets 10:20pm (Europe/Paris)

📄 🗨️ 🗨️

Limite d'utilisation atteinte • Réinitialisation 22:20 • limites partagées avec Claude Code [Continuer à travailler.](#)

Écrire un message...

+ 🗨️ Demander

Legacy Model 🗨️

Claude est une IA et peut faire des erreurs. Veuillez vérifier les réponses.

**Progression**

- ✓ Demander accès Chrome+ Finder
- ✓ Vérifier connexion Chrome MCP
- 3 Remplissage formulaire SNCF
- 4 Capturer screenshot 01 — formulaire rempli
- 5 Capturer screenshot 02 — résultats trains
- 6 Capturer screenshot 03 — récap train sélectionné
- 7 Vérifier les 3 fichiers PNG sur le Desktop
- 8 Correction des champs

**Dossier de travail**

**Contexte**

Connecteurs

- Chrome — Browser 1
- computer-use

Mai 2026

Verdict : pas pour la prod

**45× plus cher en tokens qu'une API structurée.** La boucle screenshot, analyse, action paie chaque pixel vu, à chaque étape.

**78 % de succès sur OSWorld.** Trois étapes cassées sur dix ne font pas un workflow utilisable. « Too fragile and not smart enough » (Ethan Mollick, janvier 2025).

**Notre parti pris.** OK pour des tâches ponctuelles sur app native. Pour le reste, à isoler sur machine dédiée via Dispatch, jamais sur ton poste de travail.

# Tips concrets pour s'y mettre

Ce qu'on aurait aimé savoir au jour 1



## HARNAIS

### Claude.ai et Mobile

**Projects** : un dossier Claude par contexte, custom instructions et fichiers attachés.  
Le réflexe avant de relancer la même conversation pour la centième fois.

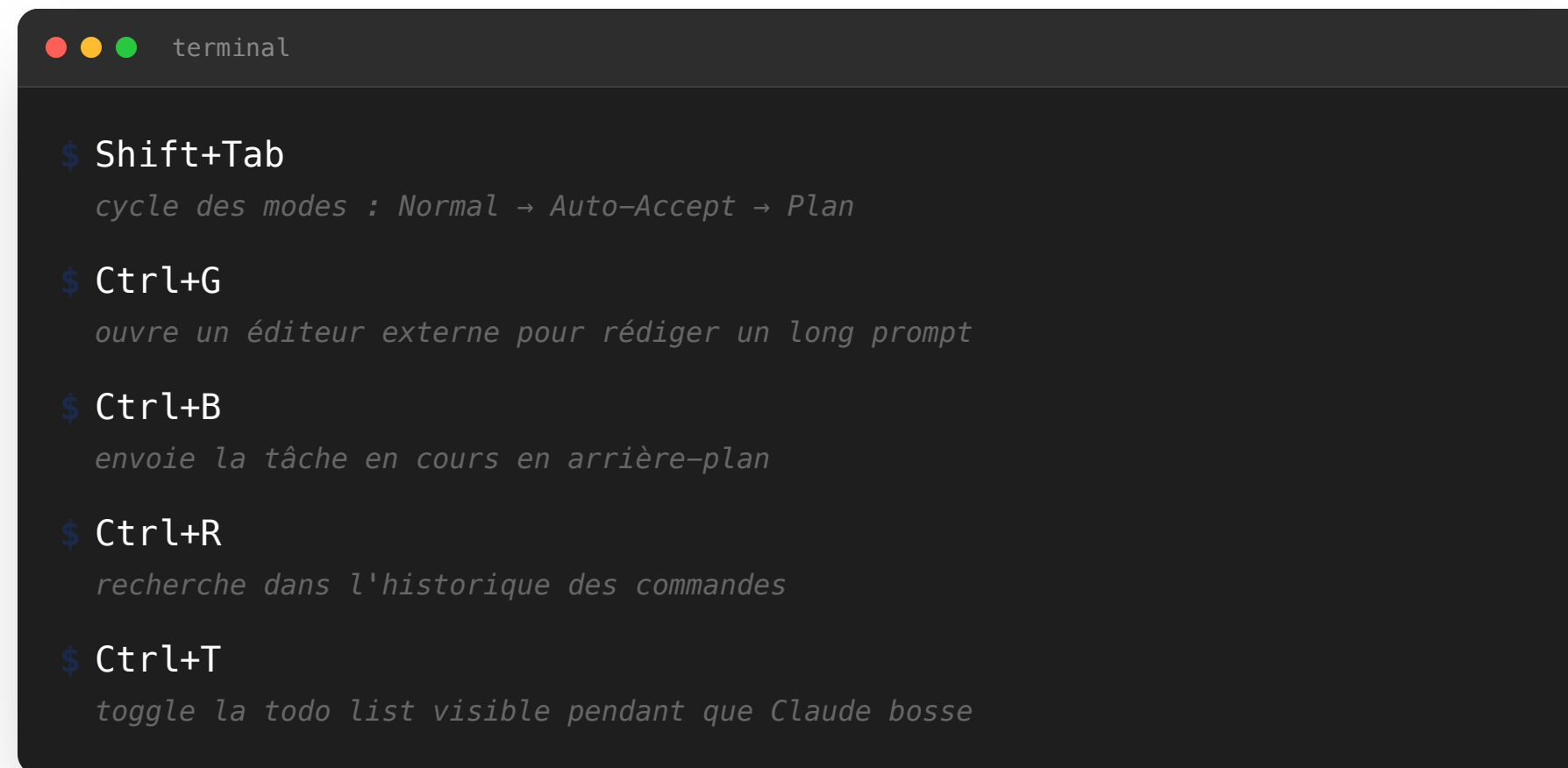
**`/schedule` dans Cowork** : la slash command pour planifier des tâches récurrentes. Ton agent reprend la main toutes les heures, tous les matins, sans toi.

**Dispatch (panneau latéral Cowork)** : tu envoies une tâche depuis ton mobile ou ton navigateur, ton Claude Code la ramasse sur ton poste de travail.

**Voice mode mobile** (anglais uniquement en mai 2026) : pour les conversations longues en main libre. Le français passe par Dictation, dispo dans 12 langues.

## HARNAIS

# Claude Code, les raccourcis

A terminal window with a dark background and light text. The title bar shows three colored dots (red, yellow, green) and the word 'terminal'. The content lists five shortcuts, each preceded by a blue '\$' symbol. The descriptions are in a light gray, italicized font.

```
terminal

$ Shift+Tab
  cycle des modes : Normal → Auto-Accept → Plan

$ Ctrl+G
  ouvre un éditeur externe pour rédiger un long prompt

$ Ctrl+B
  envoie la tâche en cours en arrière-plan

$ Ctrl+R
  recherche dans l'historique des commandes

$ Ctrl+T
  toggle la todo list visible pendant que Claude bosse
```

Sur Mac, Ctrl+V (et non Cmd+V) pour coller une image, le terminal intercepte Cmd+V avant Claude Code.

## HARNAIS

# Claude Code, les workflows

```
terminal

$ @chemin/fichier.ts
  référence directe au fichier dans le prompt, tab-completion

$ !git status
  exécute le shell et injecte le résultat dans le contexte

$ /loop 5m "check the build"
  tâche récurrente locale, expire après 7 jours

$ claude --worktree
  session isolée dans un git worktree, parfait pour paralléliser

$ Stop hook → bun test
  relance les tests à la fin, redémarre Claude si rouge
```

Plus les hooks `.claude/settings.json` (`PreToolUse`, `PostToolUse`, `SessionStart`) pour automatiser :  
format après edit, blocage `rm -rf`, contexte injecté au démarrage.

## HARNAIS

### Claude Desktop

**MCP créatifs officiels (8 connectors, avril 2026)** : Ableton, Adobe, Affinity, Autodesk Fusion, Blender, Resolume, SketchUp, Splice. Claude pilote tes outils de création depuis le chat.

**Filesystem MCP** : Claude lit et écrit dans tes dossiers locaux après configuration dans ``claude_desktop_config.json``. Le bureau devient son espace de travail.

**Le bon usage** : c'est un sandbox créatif et un assistant d'ordi, pas un IDE. Pour coder, va sur Claude Code.

## CROSS-HARNAIS

### Le pack étendable

**Skills officiels Anthropic** : `pptx`, `xlsx`, `docx`, `pdf`, `claude-api`. Plus une quinzaine d'autres (brand-guidelines, frontend-design, theme-factory, etc.). Source : [`github.com/anthropics/skills`](https://github.com/anthropics/skills).

**Connecteurs MCP** : Notion, Linear, GitHub, Postgres, Stripe et 200 autres. Listing dans Claude Desktop via l'Anthropic Connectors Directory.

**Plugins Claude Code** : skills + commands + hooks + MCP servers + agents en un bundle installable. Le pack officiel : [`github.com/anthropics/claude-plugins-official`](https://github.com/anthropics/claude-plugins-official).

**Le réflexe qui change tout** : à chaque erreur corrigée, tu ajoutes une règle dans [`CLAUDE.md`](#). Au bout de trois mois, ton agent connaît tes goûts.

## Ce qui est encore bancal

**Cowork Scheduled Tasks affiche « 1 heure minimum ».** L'API derrière accepte sans broncher une cron toutes les 3 minutes. Bride d'UX, ou roadmap pas alignée entre l'UI et l'API.

**Claude Code retiré du Pro plan sans annonce, le 21 avril 2026.** Aucun email, aucun changelog, juste une mise à jour de doc. Reversé en 24 heures après backlash. Anthropic n'a jamais expliqué clairement.

**Les Skills dans `~/.claude/skills/` chargent le fichier entier,** pas juste le frontmatter. 10 000 tokens consommés au lieu de 150, silencieusement, à chaque session. Comportement opposé aux project skills, non documenté.

# Place à la démo.

Cowork en usage réel



## Trois choses à retenir

**Un seul modèle, plusieurs harnais.** Choisir le bon harnais pour le bon usage compte plus que choisir le bon prompt.

**L'écosystème bouge vite, et certaines décisions sont encore en train d'être prises.** Ce qui est vrai aujourd'hui peut changer la semaine prochaine.

**Le bon réflexe : essayer petit, vite, sur un cas réel.** Pas besoin d'attendre la version définitive. La version définitive n'arrivera jamais.

# À vous.

Questions, retours, contradictions ?

