

生成AIによる プログラミングの始め方

第3回：もっと安くAIコーディング

大阪大学D3センター 谷口昂平

セミナーの内容

第1回：9/10 10:00 - 12:00	ご挨拶 大阪大学D3センター 先進高性能計算基盤システム研究部門 伊達 進
	AIコーディング入門：座学（デモあり） AIコーディングの歴史、LLMモデルとハーネス、Cursorでスパコンに接続～ジョブ実行スキル作成 (講師：大阪大学D3センター 先進高性能計算基盤システム研究部門 / 大阪大学大学院 情報科学研究科 谷口 昂平)
第2回：9/17 10:00 - 12:00	もっと賢くAIコーディング：座学 プランモード、サブエージェント、ループエンジニアリング (講師：大阪大学D3センター 先進高性能計算基盤システム研究部門 / 大阪大学大学院 情報科学研究科 谷口 昂平)
第3回：9/24 10:00 - 12:00 今回	もっと安くAIコーディング：座学 トークン節約テクニック、ローカルLLM、無料モデル（OpenCodeなど） ※利用可能なモデル等の状況により、内容を変更する場合があります。 (講師：大阪大学D3センター 先進高性能計算基盤システム研究部門 / 大阪大学大学院 情報科学研究科 谷口 昂平)

第1回

AI コーディングを
とりあえず使える

第2回

AI コーディングで
難しいものを作る

第3回

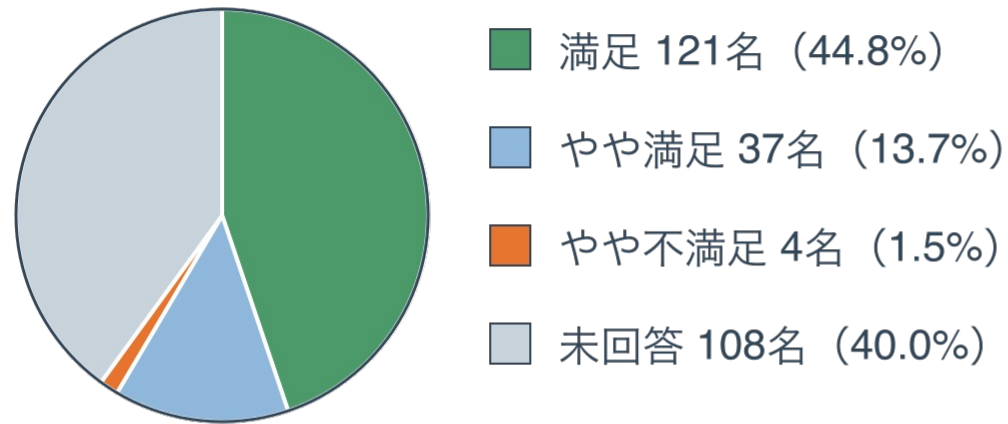
AI コーディングで
安く作れる

第2回アンケートの結果

出典：第2回受講者アンケート（2026.9）

参加者270名のうち、163名が回答
具体性と、講義の長さ・テンポへの指摘も

参加者270名の満足度



満足度の回答 162名 (60.0%)

自由記述から

参考になった点

たとえ話・実体験・最新の情報

具体性・始め方

実際の操作や最初に取り組み方

講義の長さ・テンポ

「もう少し短くてもいいかもしれない」

「録画の1.5倍速くらいとか。」

シルバーウィーク, お金使いました？

出典：ぐるなび「シルバーウィーク調査」(2026.9)

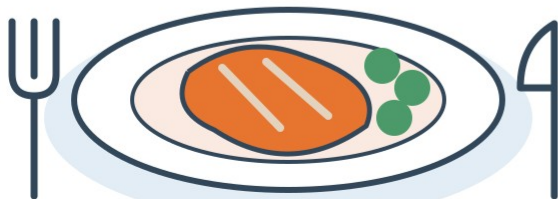
ぐるなびの連休前調査
いちばん高い外食・1食・1人

外食にかかる平均予算

6,094円

予定予算／回答346人

連休のごちそう



AIの利用料は
抑えたい...



連休明けのAIコーディング

連休中に登場したAIモデル

出典：[OpenAI \(9/22\)](#)

[Anthropic \(9/22\)](#)

[xAI \(9/21\)](#)

[Xiaomi \(9/22\)](#)

安く使える新モデルが増えた。各社の狙いは？

発表日・開発元

モデル

発表で重視した改善

9/21 xAI

Grok 4.7

単価を据え置き，長い仕事の
粘り強さ・自己検証を改善

9/22 Xiaomi

MiMo-V2.6-Pro・Flash

少ない手順・トークンで
仕事を終える効率

9/22 Anthropic

Claude Opus 5.5

標準設定の典型的な仕事で
旧Opus 5より費用40%減

9/22 OpenAI

GPT-6 Sol・Luna (Codex)

推論とキャッシュを効率化し，
安く・長く使えるように

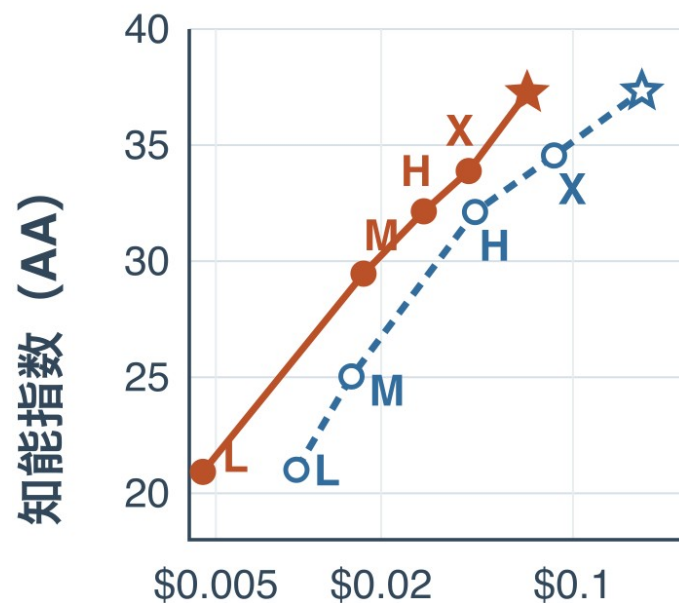
OpenAIとAnthropicの新旧モデル

出典：Artificial Analysis「知能指数・一課題あたり費用」（2026.9）

新旧モデルの性能と，1課題あたりの費用

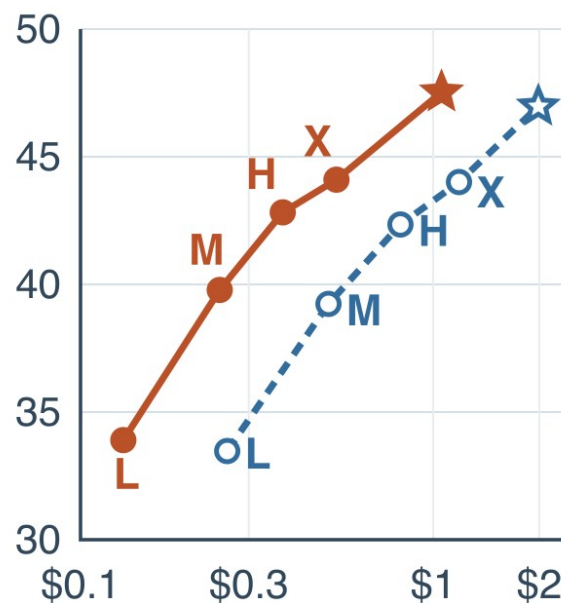
GPT Luna (OpenAI)

--- 旧 5.6 — 新 6



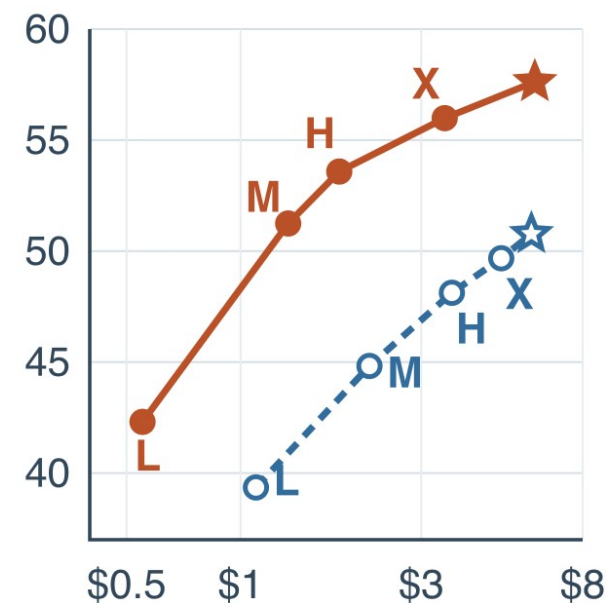
GPT Sol (OpenAI)

--- 旧 5.6 — 新 6



Claude Opus (Anthropic)

--- 旧 5 — 新 5.5



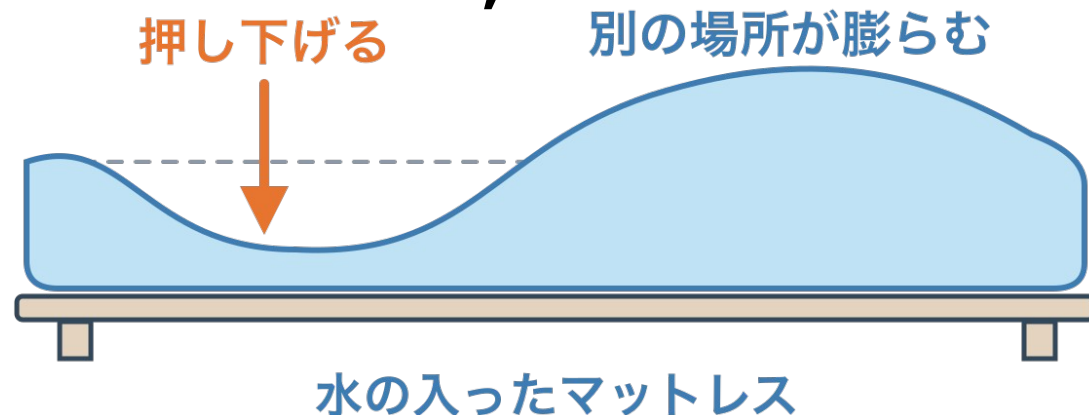
一課題あたり費用 (米ドル・対数軸) ← 安い

L : low M : medium H : high X : xhigh ★ : max

ウォーターベッド効果

出典：Inderst・Valletti「Buyer Power and the Waterbed Effect」(2011)

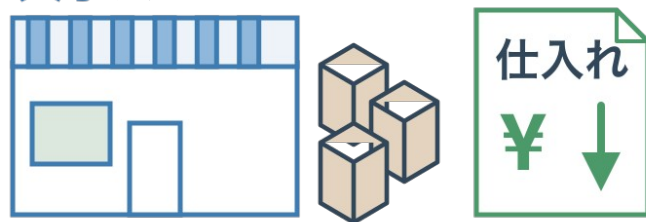
同じ地域で競う，大手スーパーと小さな店の例



一方の値下げが，
別の取引での値上げに
つながることがある

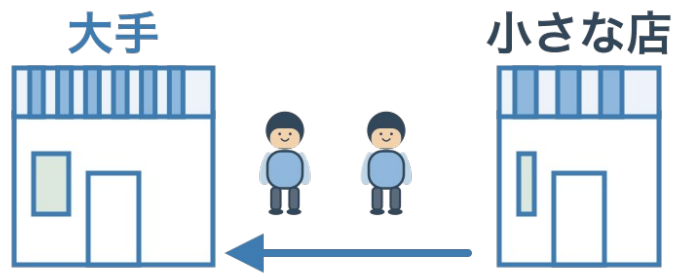
① 大手が安く仕入れる

大手スーパー



まとめ買いを武器に
値引きを引き出す

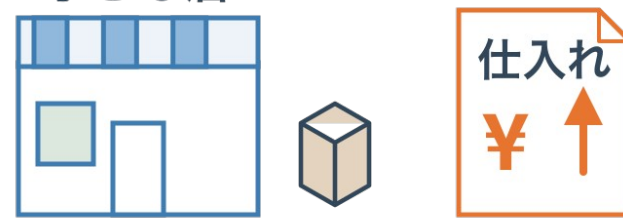
② 大手へ客が移る



店頭でも値下げすると
小さな店の客も大手へ

③ 小さな店の仕入量が減る

小さな店



値引き交渉が不利になり
仕入価格が上がることも

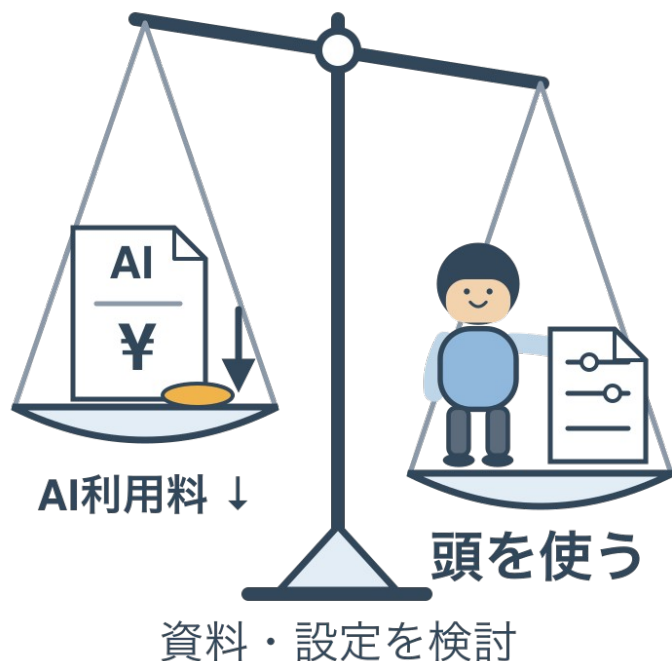
AI利用料と引き換えに負担するもの

出典：OpenCode「Zen / Privacy」(2026.9)

利用料を下げる代わりに、何を引き受けるか

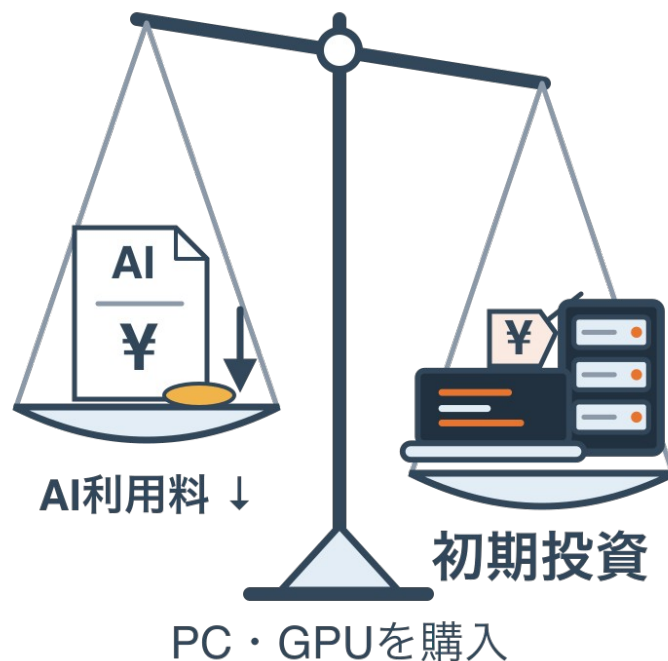
トークン節約

課金ルールと手順を工夫



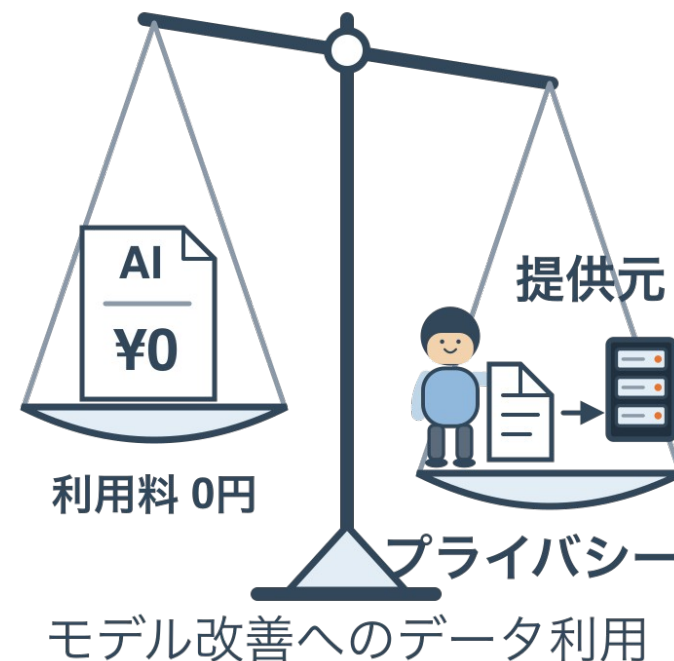
ローカルLLM

手元の機材で繰り返す



無料モデル

データ利用が条件の無料版



今日のお品書き

1. トークン節約テクニック
頭を使って，余分な支払いを減らす
2. ローカルLLM
設備を先に用意し，手元で繰り返し使う
3. 無料モデル
学習利用が条件なら，代償はプライバシー
4. この先のAIコーディング
料金と品質の変化に備え，使い続ける方法を考える

今日のお品書き

1. トークン節約テクニック
頭を使って，余分な支払いを減らす
2. ローカルLLM
設備を先に用意し，手元で繰り返し使う
3. 無料モデル
学習利用が条件なら，代償はプライバシー
4. この先のAIコーディング
料金と品質の変化に備え，使い続ける方法を考える

トークンとは

出典：OpenAI「tiktoken」（2026.9計数）

「GPT-4」（2023.3）

「GPT-4o」（2024.5）

AIは文章を「トークン」という単位で扱う

同じ文：今日はいいい天気です。

10文字

GPT-4（2023年公開）

9トークン

今 日 は い い 天 気 です 。

1枠が1トークン

GPT-4o（2024年公開）

後継モデル

6トークン

今日は いい 天 気 です 。

「今日は」は3個から1個，「いい」は2個から1個へ

コード修正一件のAI利用料

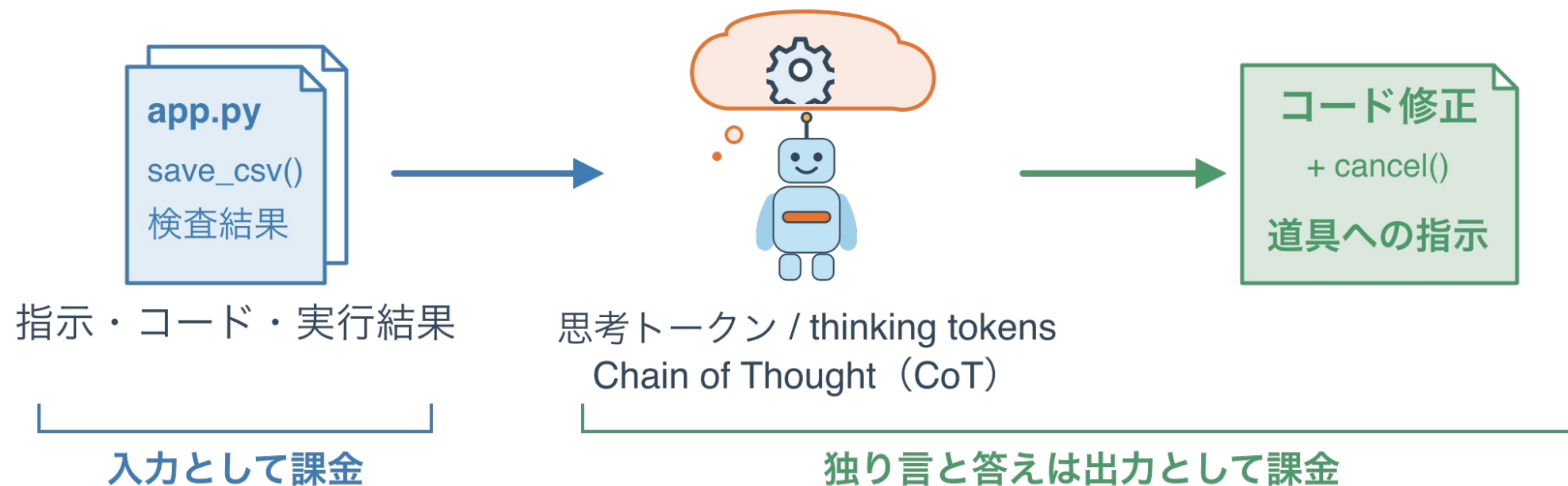
出典：Anthropic「Pricing」（2026.9）

四つの費用：読んだ分は入力，独り言と答えは出力として課金

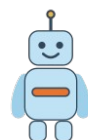
① 読んだトークン

② 独り言トークン

③ 答えのトークン



④ 付帯コスト



補助AI

推薦・会話名・メモリー



有料ツール

外部サービスの処理料

翻訳税：日英の文字数とトークン数

出典：国連「世界人権宣言」

Qwen「Qwen3.8-27B」（2026.9）

Ahiaほか（2023.12）

同じ意味の文章を，日本語と英語で比べる

世界人権宣言 第1条の冒頭部分

一枠が1トークン（Qwen3.8-27B）

英語

All human beings are born free and equal in dignity and rights

日本語

すべての人間は、生れながらにして自由であり、かつ、
尊厳と権利とについて平等である

文字数（空白を含む）

英語

62

日本語

40

トークン数（同じモデル）

英語

12

日本語

21

翻訳税：五言語のトークン数

出典：国連「世界人権宣言」

Qwen「Qwen3.8-27B」(2026.9)

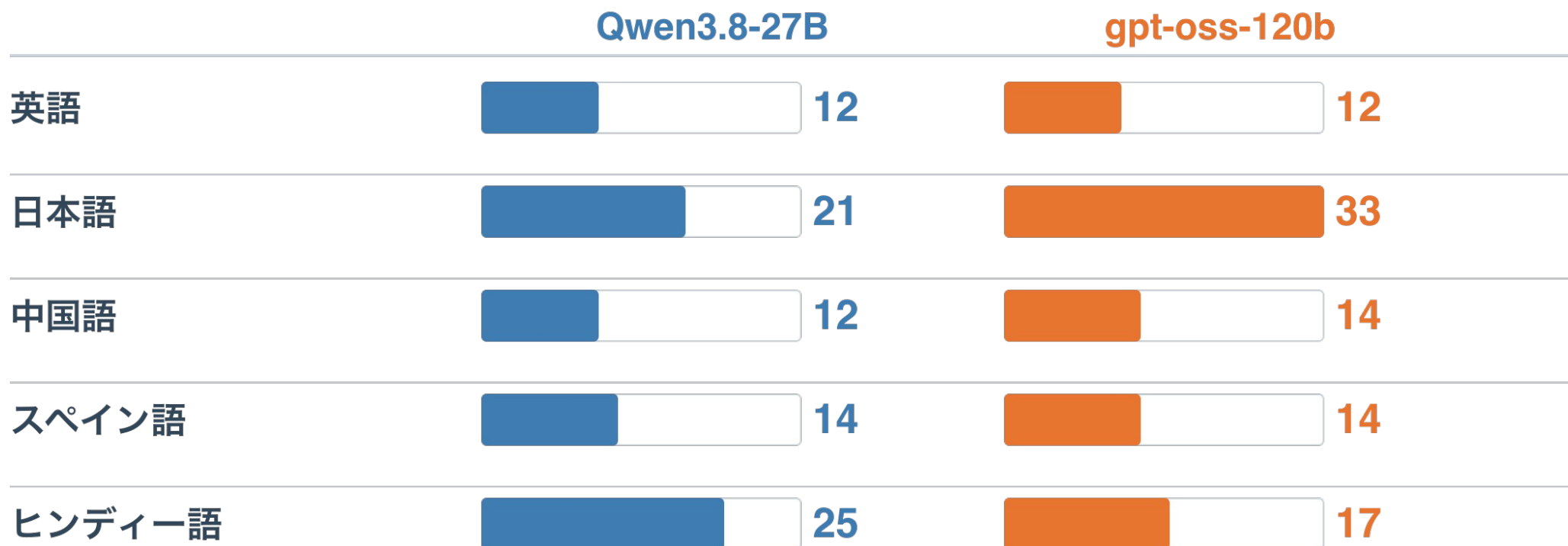
OpenAI「gpt-oss」(2025.8)

同じ意味の文章を，二つのモデルの区切り方で数える

同じ内容でも，トークナイザーで差の出方が変わる

世界人権宣言 第1条の冒頭部分・文末句読点なし

入力トークン数



読んだトークンの節約

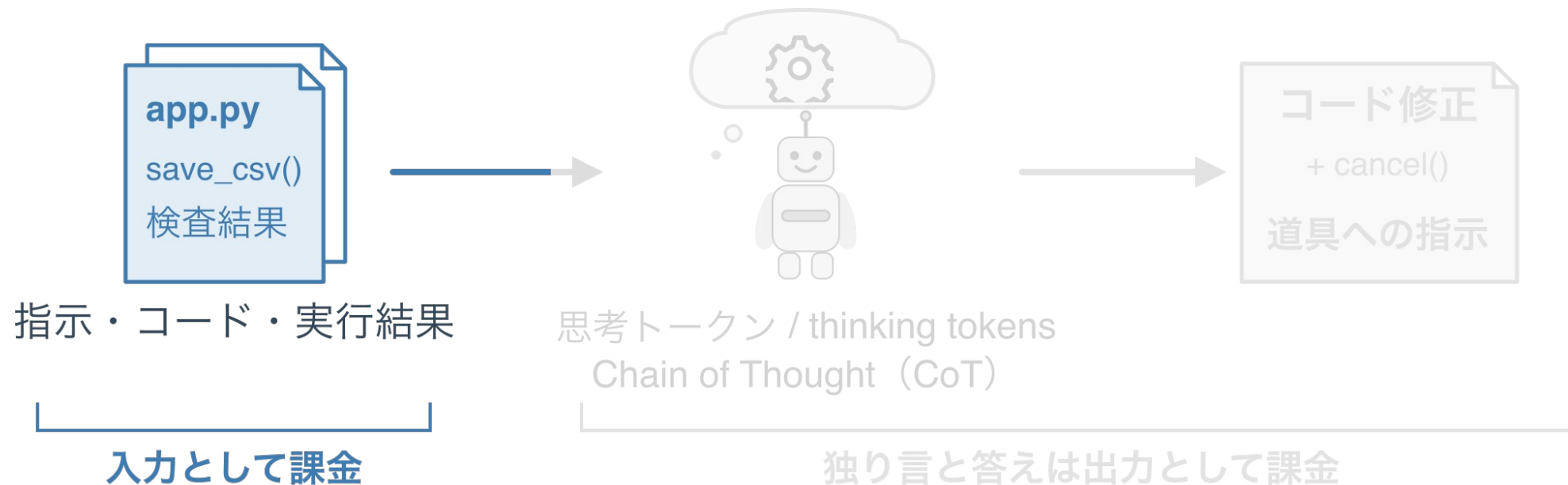
出典：Anthropic「Pricing」（2026.9）

必要な資料を選び，同じ入力の計算を再利用する

① 読んだトークン

② 独り言トークン

③ 答えのトークン



④ 付帯コスト



補助AI

推薦・会話名・メモリー



有料ツール

外部サービスの処理料

スキルと道具の説明の読み込み

出典：Anthropic「Skills」

Anthropic「道具検索」

OpenAI「Build skills」

一覧から選び、必要な手順や説明を後から読む

既習：スキルを使う条件

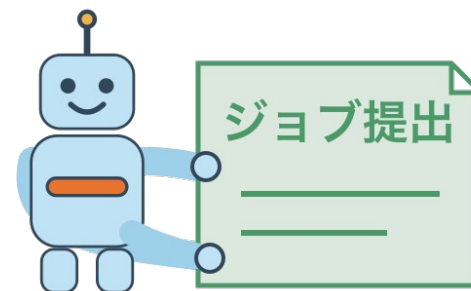


全てを最初から持たせると...



使わない手順も入力に入る

仕事に合う手順書だけ持つ



今回読む
量を絞る

道具検索も：必要な定義だけ後から読む

プロンプトキャッシュとは

出典：[OpenAI キャッシュ・料金](#)

[Codex料金](#)

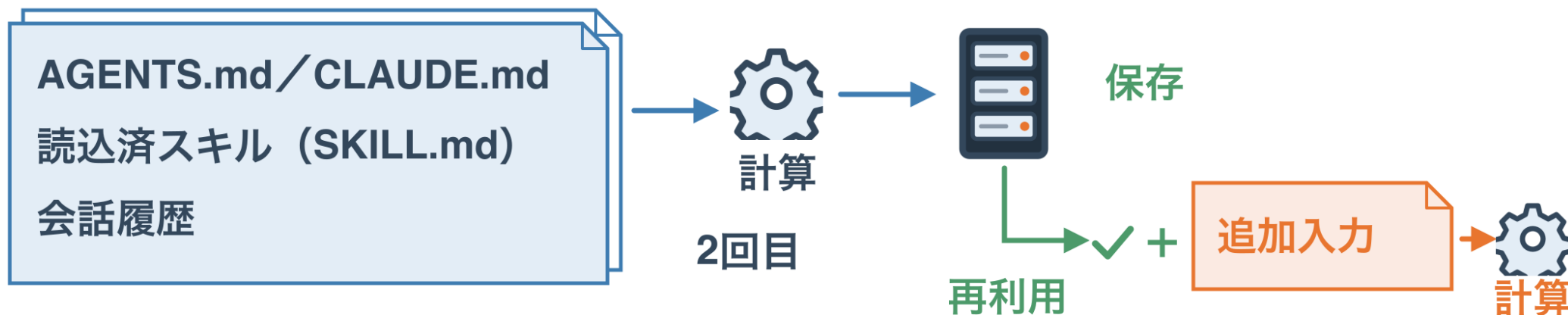
[Claude Code 保持時間](#)

[Claude API料金](#)

同じ入力の計算を再利用し，入力を安く・速く処理する

共通する入力の前半

1回目



保持時間と，ヒットした入力の単価

OpenAI

GPT-6 API：最後の利用から最低30分

Codex追加クレジット：

ヒット入力の単価1/10

Anthropic

Claude Code：定額内の会話は1時間

追加クレジットは既定5分

Opus 5.5 API：ヒット入力単価1/20

共通指示と作業履歴を短くする

出典：OpenAI「Astraの指示見直し」（2026.9）

Anthropic「How Claude Code works」（2026.9確認）

共通指示を整理し、長くなった作業履歴を要約する

共通指示・資料を整理する

修正・検査の履歴を要約する

AGENTS.md / CLAUDE.md

毎回、設計・DB・公開手順を読む



条件を絞る

DB変更時はDB仕様を読む
今回に必要な資料だけを渡す

読む材料そのものを減らす

UTF-8に決定

pytest：成功

app.pyを修正

次は取消ボタン

作業の続きに必要な要点

app.py：保存を実装 / 検査成功
仕様：UTF-8 / 未完了：取消

コンテキスト圧縮

道具の出力を短くするRTK

出典：RTK (Rust Token Killer) 公式README (2026.9確認)

講師の使用経験 (2026.9)

テストの成功一覧や長いログを省いてから, AIへ渡す

例：テストの生出力

RTK

AIへ返す出力

```
PASS test_save  
PASS test_cancel  
FAIL test_timeout
```

選別・省略

```
FAIL test_timeout  
成功 2 / 失敗 1
```

① 読んだトークンを減らす

成功した検査の一覧は省略

省かれた情報は, AIには見えていない



「test_cancelも実行した？」



生出力を確認

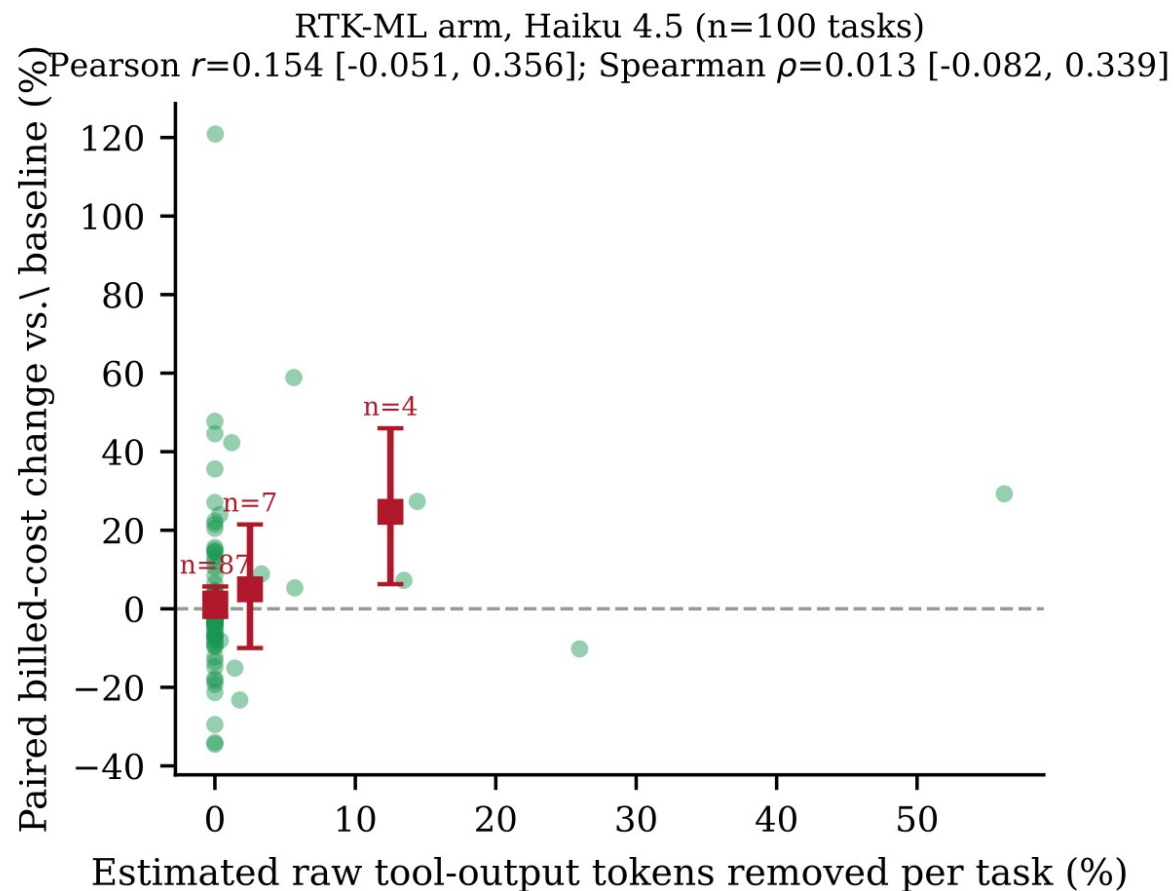
rtk proxy pytest -v

講師の経験：論文の原文を十分に読めず, 別の読取り経路へ

トークン削減と一件の利用料

出典：Weinberger・Hozez「Token Reduction Is Not Cost Reduction」（2026.8改訂）

道具の出力を圧縮したときの，削減量と利用料を比べる



横：道具の出力の削減率

縦：圧縮なしとの請求差

Claude Code：3モデル，各方式712回

RTK-ML（論文の構成）

562万→346万トークン

-38.4%（推定）

課題ごとの請求差の平均

+6.8% 圧縮なし比

平均の返答回数も増加

4.46回 → 4.69回

独り言トークンの節約

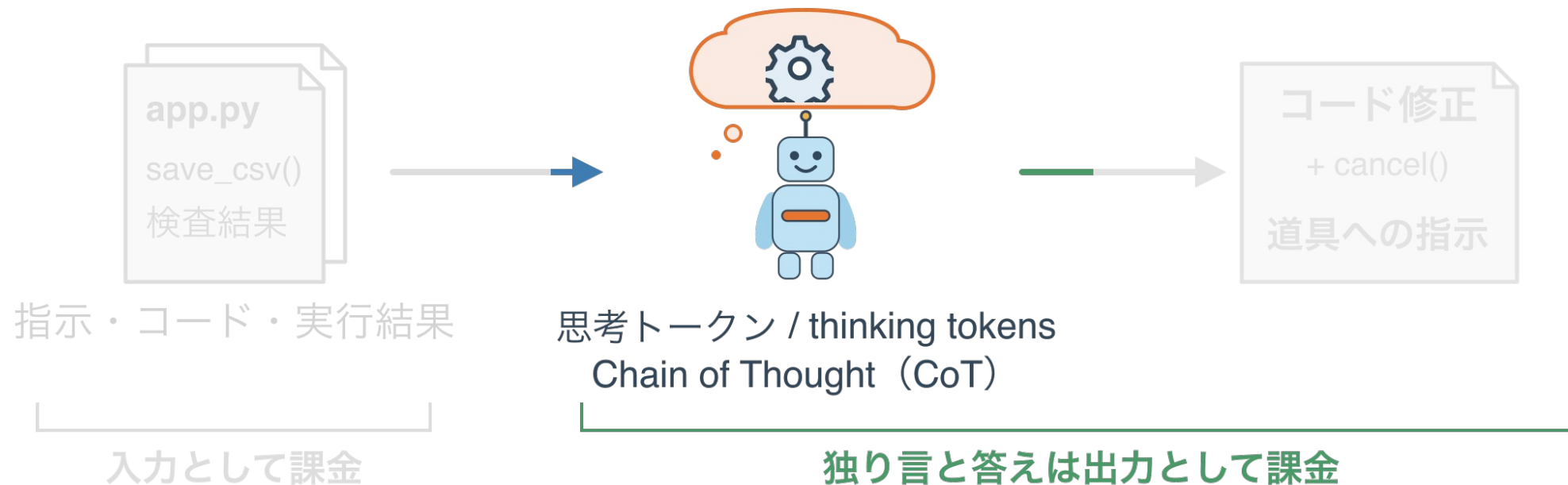
出典 : Anthropic「Pricing」 (2026.9)

effortで考える強さを選び、必要な仕事に推論を使う

① 読んだトークン

② 独り言トークン

③ 答えのトークン



④ 付帯コスト



補助AI

推薦・会話名・メモリー



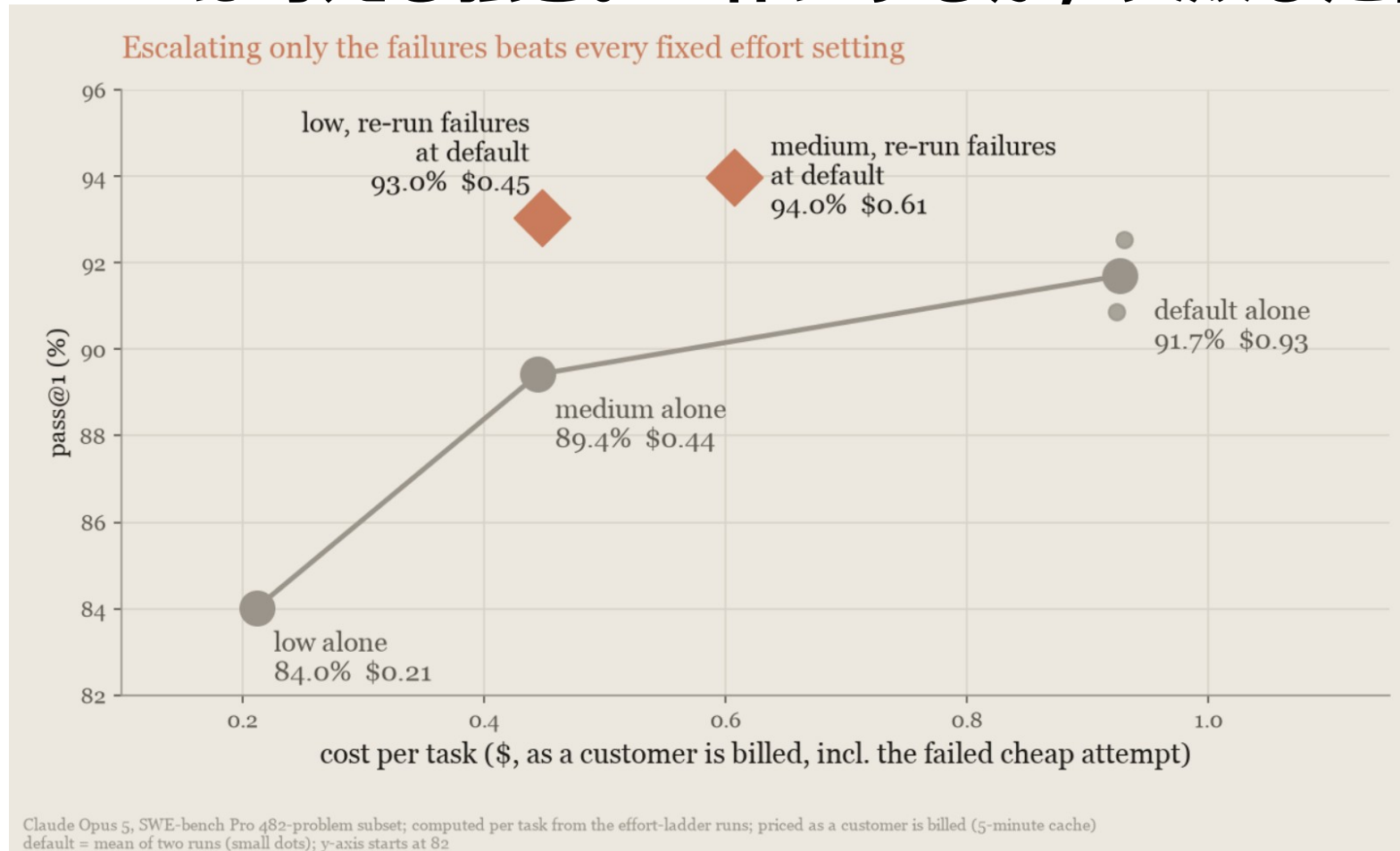
有料ツール

外部サービスの処理料

考える強さ (effort) と再試行

出典: Anthropic「Optimizing for cost and intelligence」(2026.9)

effortは考える強さ。一律にするか，失敗した仕事だけ上げるか



Opus 5

SWE-bench Pro

低effortから
失敗だけ再試行

93% / \$0.45

全件を既定

91.7% / \$0.93

失敗検知が条件

Jevによるモデル・effortの自動選択

出典：[Jevルーター](#)

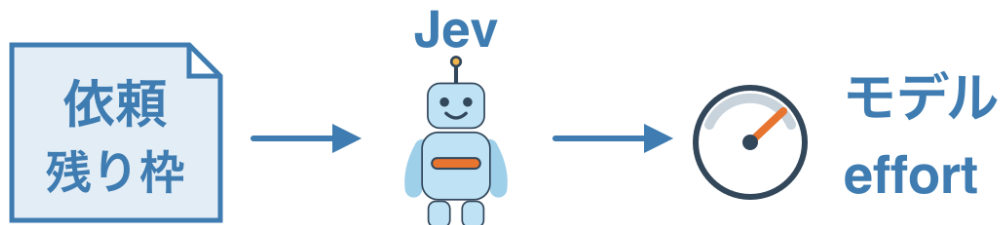
[Jevスキル版](#)

[Claude Code \(2026.9\)](#)

[Codex v0.156.1 \(2026.9\)](#)

依頼に応じた強さの選択を，スキルやツールに任せる

Jev Codex Router



スキル版

子エージェントの選択

macOS・Codex用CLI

会話で固定／毎ターン選択

同じモデルでeffortを途中変更するとき

Claude Code + Opus 5.5

v2.1.280以降

Claudeサブスクの標準設定

✓ そのままキャッシュを保持

Codex v0.156.1・GPT-6系

モデル側の対応＋実験機能ON

config.toml

```
[features]
reasoning_effort_override = true
```

答えのトークンの節約

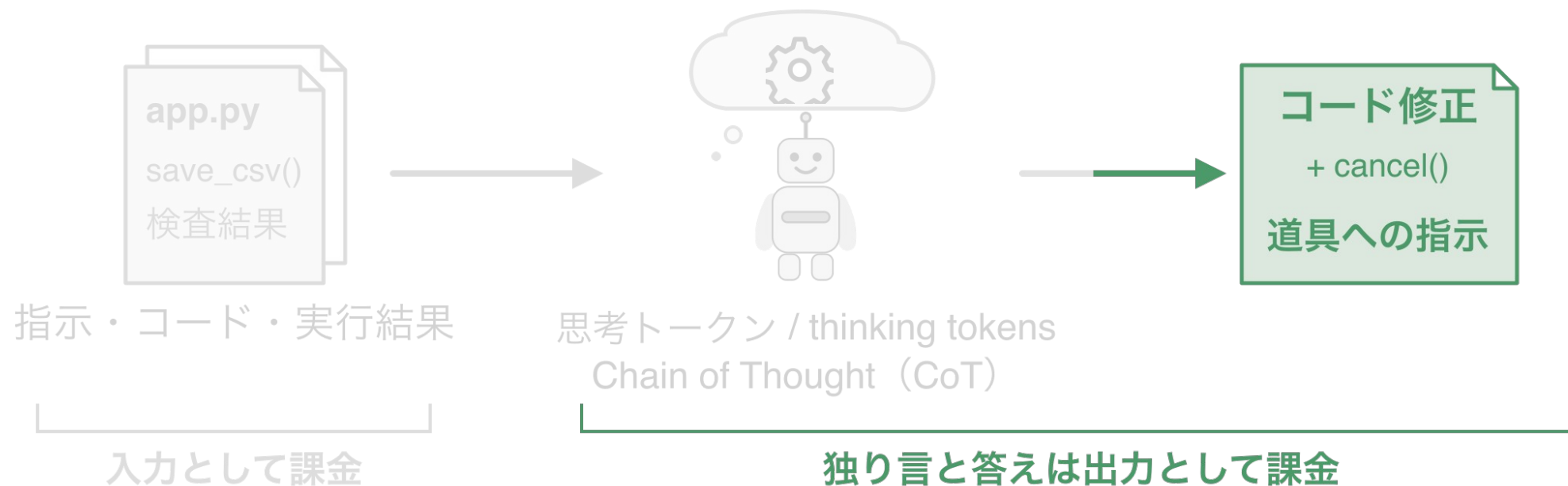
出典 : Anthropic「Pricing」 (2026.9)

コードは差分で編集し，報告では全文の再掲を減らす

① 読んだトークン

② 独り言トークン

③ 答えのトークン



④ 付帯コスト



補助AI

推薦・会話名・メモリー



有料ツール

外部サービスの処理料

差分の編集と作業後の報告

出典 : Anthropic「編集ツール」

OpenAI「Prompting」

ayghri「i-have-adhd」

待ち時間の一行を直し、変更点と検査結果を報告する

編集時：client.py の待ち時間を 30秒 → 60秒へ

変更前のファイル

```
host="example.com"  
timeout=30  
debug=False
```

全文を生成

```
host="example.com"  
timeout=60  
debug=False
```

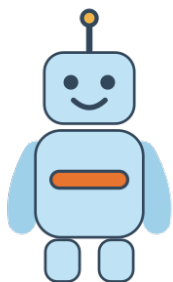
変えない2行も生成

差分を適用

```
- timeout=30  
+ timeout=60
```

置き換える箇所を指定

報告時：前置き・コード全文の再掲を省く



変更点：client.py の待ち時間を 30 → 60 秒

検査結果：構文チェック成功

未確認：サーバーとの実接続

Claude Codeの短文指示と品質低下

出典 : Anthropic「Claude Codeの品質問題」(2026.4)

返答を短くする指示を追加・撤回した経緯と評価結果

Anthropicが公開した短文指示の原文

4月16日 : Claude Code のシステム指示に追加

“Length limits: keep text between tool calls to ≤ 25 words. Keep final responses to ≤ 100 words unless the task requires more detail.”

途中の説明 : 25語以下 最後の返答 : 原則100語以下

追加のコーディング評価 1 件

Opus 4.6・4.7 とともに

3%低下

4月20日 : 短文指示を撤回

付帯コスト：裏で動くAI

出典：Claude Code「推薦」（2026.9）

OpenCode「設定」（2026.9）

Codex「メモリー」（2026.9）

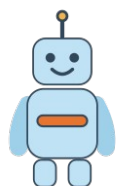
推薦・会話名・メモリーにもAIを使う

本作業のAI

④ 付帯コスト：裏で動く補助AI

① 読んだトークン

指示・コード・実行結果



② 独り言

修正方法を考える

③ 答えのトークン

コード修正・道具への指示

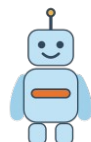
Claude Code



次のプロンプトを推薦

「テストも追加しますか？」

OpenCode



会話名を作る

CSV保存処理の修正

Codex



メモリーを作る

好み・過去の作業を整理

付帯コスト：有料ツールへの委託

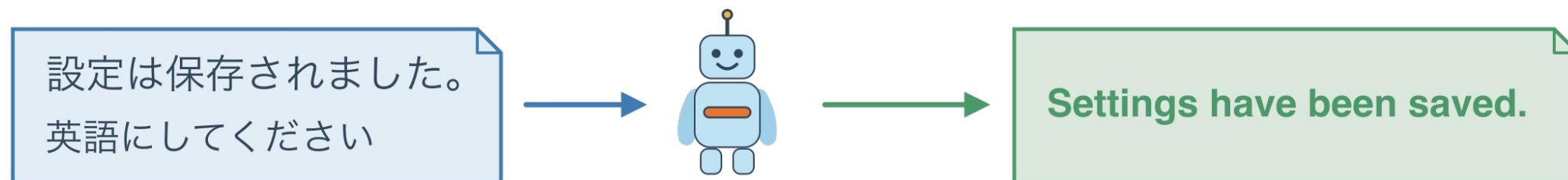
出典：DeepL「公式MCP」(2026.9)

DeepL「APIの利用量と請求」(2026.9)

同じ英訳でも，処理を任せる先で請求が変わる

例：READMEの一文を英訳

モデルが訳文を書く



本作業の①入力・②独り言・③出力

訳文作成を有料ツールへ委ねる



ハーネスで変わる利用料

出典：Panほか「HarnessTax」（2026.9）

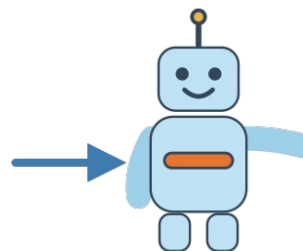
同じモデル・同じ開発課題でも，アプリで差が出る

同じ Fable 5・同じ開発課題で比べる

アプリが用意する入力

共通指示 → 手順・道具の仕様

実行結果 → 次の要求の履歴



同じモデル

修正を試す

同じ開発課題30件 × 各3回

使うアプリ	成功率	トークン料金／試行
Claude Code	97.8%	\$1.33
Codex	96.7%	\$0.89
Pi（軽量ハーネス）	96.7%	\$0.67

まとめ：トークン節約テクニック

出典：講師の利用方針（2026.9）

[i-have-adhd](#)

必要な推論・修正・検査を残し，上乘せを減らす

指示・追加機能を定期点検 モデルの切替

effortの変更

報告を簡潔に

作業の区切りに

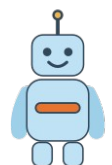
対応環境で必要時調整

i-have-adhd程度に



裏で動くAIも点検

推薦・会話名・メモリー



不要な補助処理を減らす

有料ツールへの委託も見直す

今日のお品書き

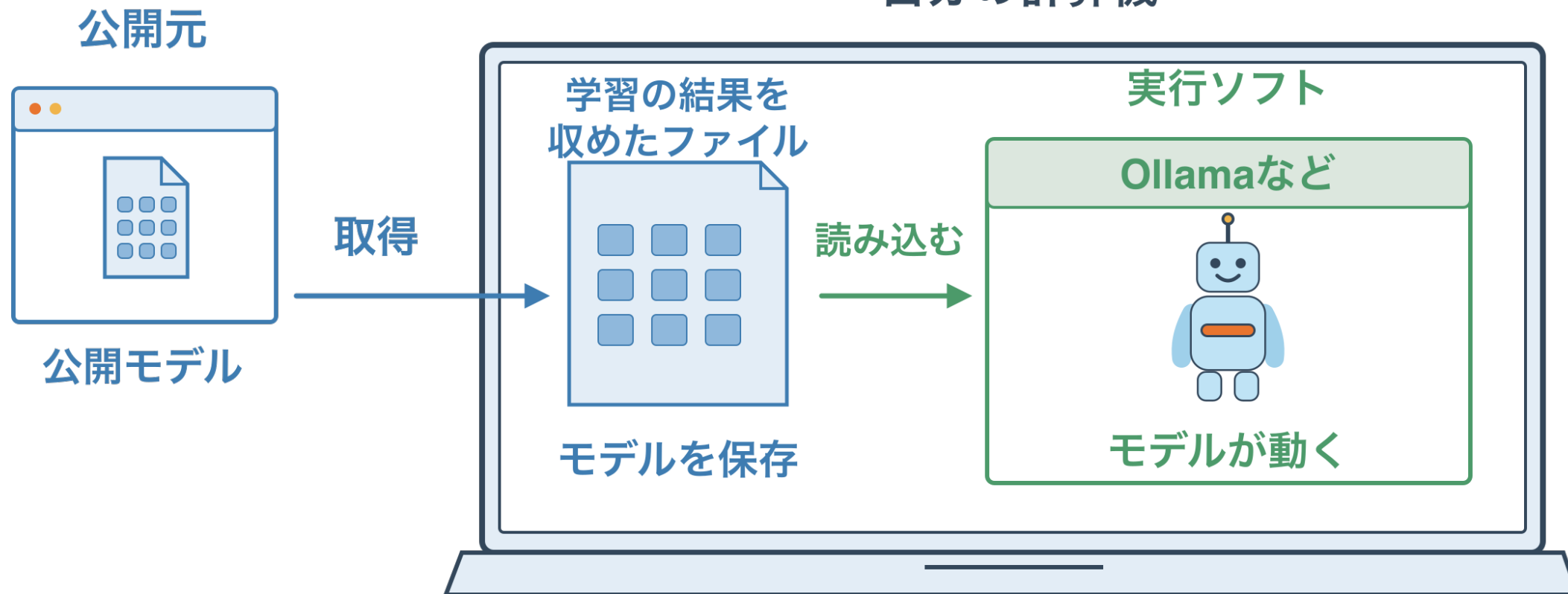
1. トークン節約テクニック
頭を使って，余分な支払いを減らす
2. ローカルLLM
設備を先に用意し，手元で繰り返し使う
3. 無料モデル
学習利用が条件なら，代償はプライバシー
4. この先のAIコーディング
料金と品質の変化に備え，使い続ける方法を考える

ローカルLLMとは

出典：Ollama公式ドキュメント

学習済みのモデルを，自分の計算機で動かす

自分の計算機



ローカルLLMの歴史（2023～2024年）

出典：[llama.cpp（2023.3）](#)

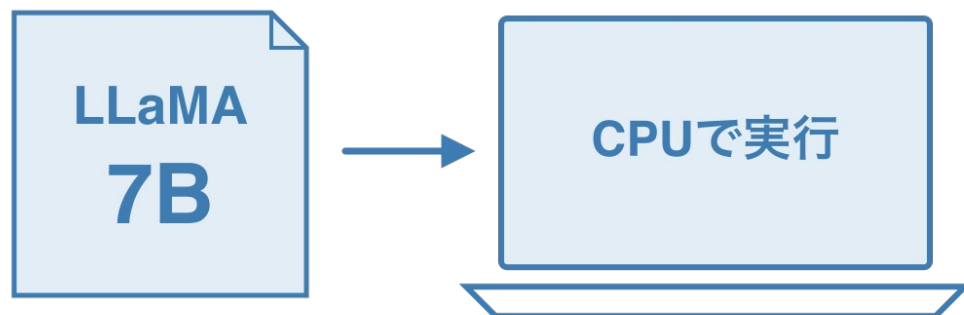
[Meta「Llama 3.1」（2024.7）](#)

[Supermicro「H100構成」（2024.9）](#)

公開モデルの大型化と、増える機材の負担

2023.3 llama.cpp

7B=70億パラメータ



4ビットの重み：約4GB

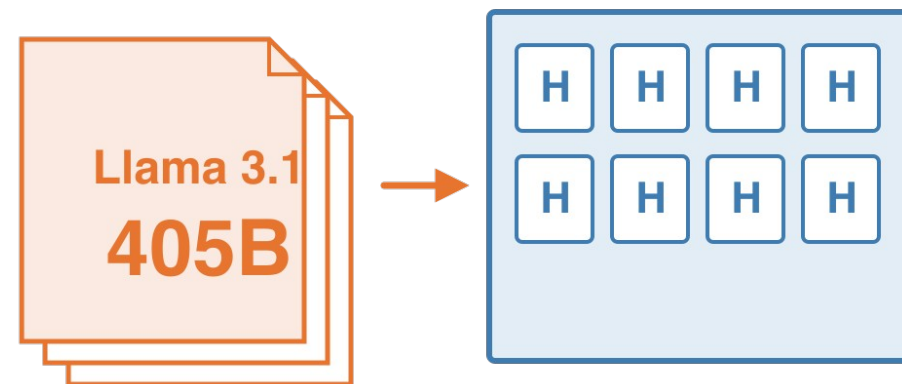
M1 Pro MacBook Pro

ノートPCのCPUで実行

重みの容量とは別に、会話と計算に使うメモリも必要

2024.7 Llama 3.1

405B=4050億パラメータ



8ビットの重み：概算405GB

H100 SXM × 8

GPUを8基積んだサーバー

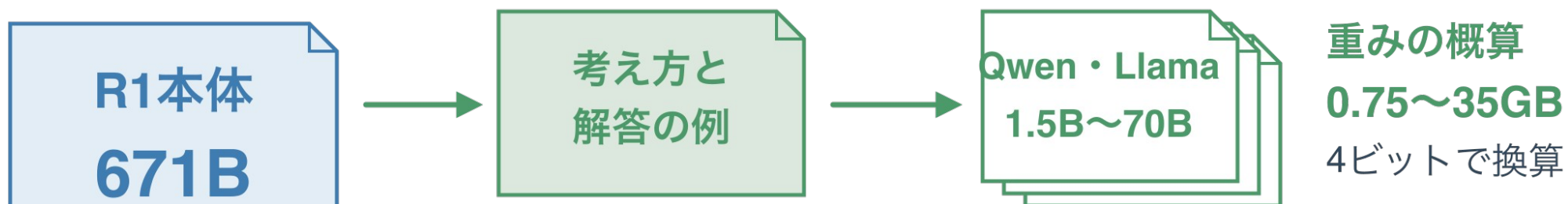
DeepSeek-R1の知識蒸留と量子化

出典：DeepSeek「R1」（2025.1） ifioravanti「R1」（2025.3） Geerling「Mac Studio」

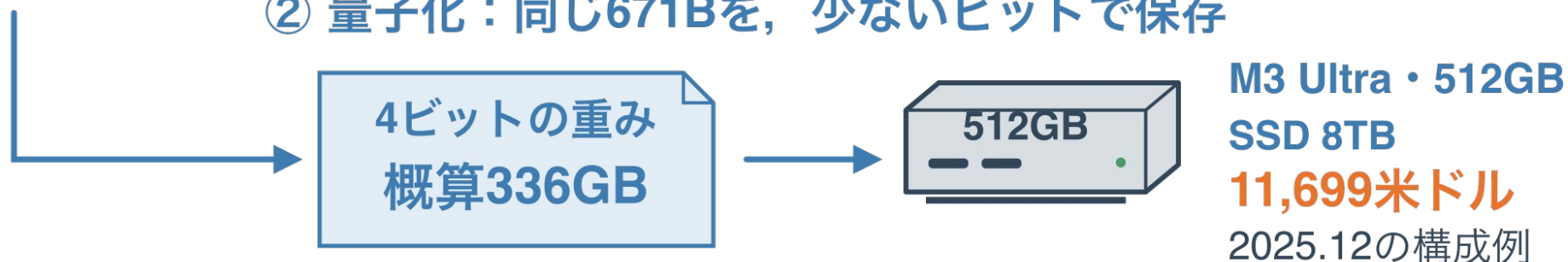
知識蒸留と量子化で，何を小さくするかが違う

2025.1 DeepSeek-R1：本体は671B（6710億パラメータ）

① 知識蒸留：解答例から，別の小型モデルを学習



② 量子化：同じ671Bを，少ないビットで保存



重みのみの単純換算。実行には会話・計算用のメモリも必要

ローカルLLMの歴史（2025～2026年）

出典：OpenAI「gpt-oss」（2025.8）

DeepSeek「0731」（2026.7）

Classmethod（2026.8）

大容量の共有メモリを使うPCにも，実行の場が広がる

2025.8 gpt-oss

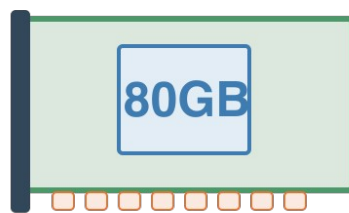
動かすメモリと推論能力を示して公開

gpt-oss-20b 16GB

主要な推論評価でo3-miniに近い

gpt-oss-120b 80GB

主要な推論評価でo4-miniに近い



H100 80GB

GPU 1基で動く設計

2026.7.31 DeepSeek

V4 Flash 0731

コード編集・道具の操作が進歩

公式FP8の重み 167GB

128GBの1台には入らない



DGX Spark 2台に分ける実行例

GPU専用メモリと共有メモリ

出典：vLLM「Offload」 (2026.9)

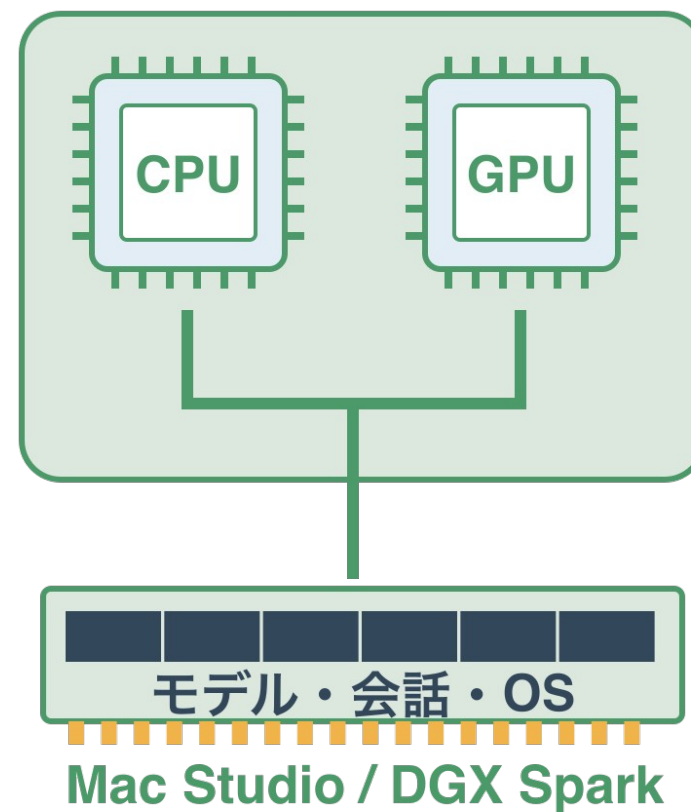
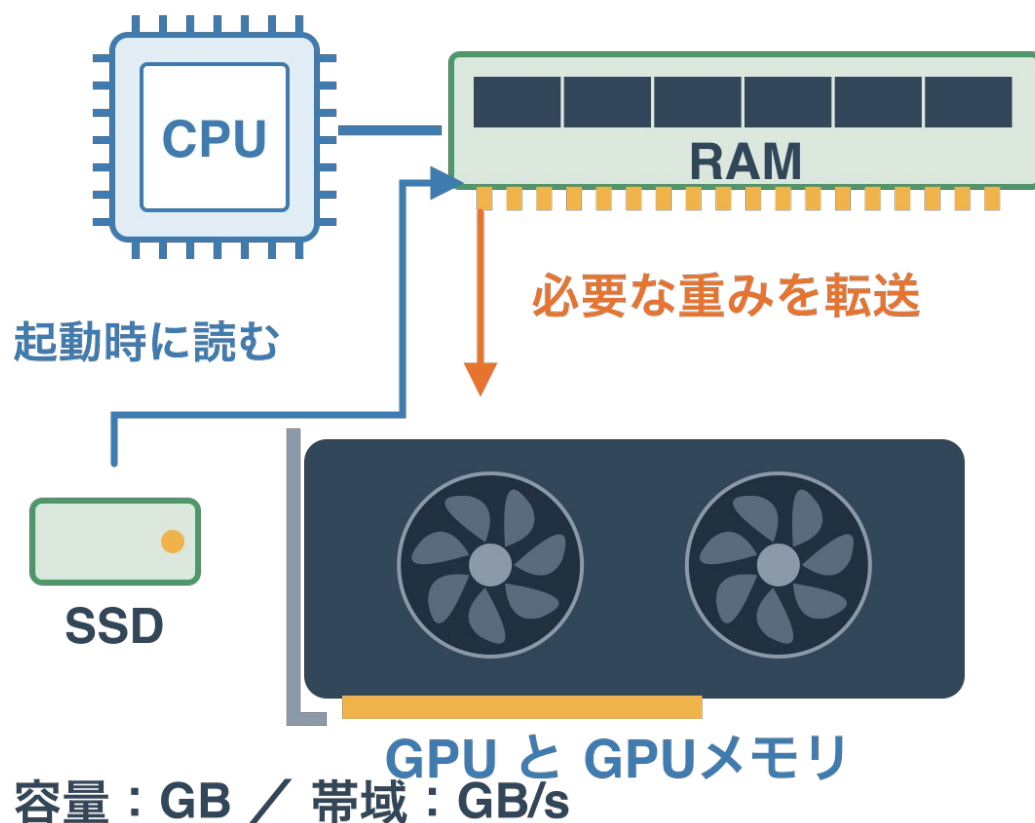
NVIDIA「Local AI」 (2026.9)

Apple「Mac Studio」 (2026.9)

重みを置く場所と，CPU・GPU間での移動を比べる

独立GPU：主記憶とGPUメモリが分かれる

共有メモリ：同じ領域を使う



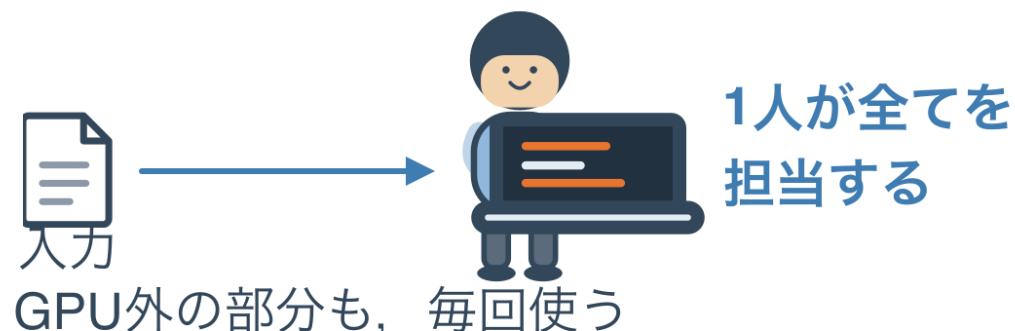
DenseとMoEの計算・メモリ

出典：Lon Seidman「V100実行記録」（2026.9）

FreeToken論文（2026.8）

計算の仕組みと、公開された実行構成の例

Dense：全体を毎回使う



MoE：使うExpertを選ぶ

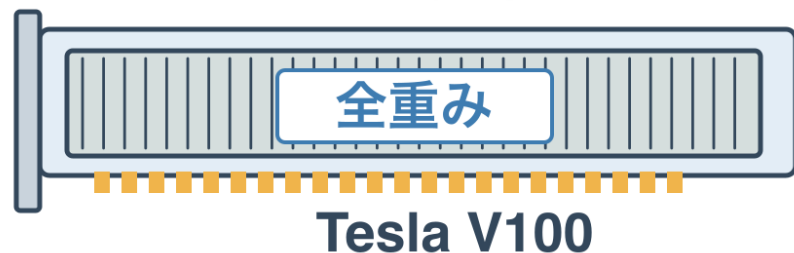
Expert＝分担する計算部分



Dense：Qwen3.8-27B

Q4_K_M／全層をGPUに配置

VRAM 32GB × 1

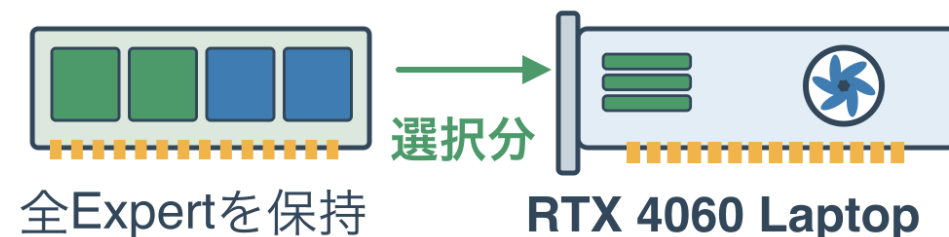


MoE：Qwen3.6-35B-A3B

NVFP4／FreeToken

本体RAM 32GiB

VRAM 8GB × 1



歴史

使う理由

快適さの条件

購入構成

講師が使うローカルLLM

出典：講師の使用経験・実機構成（2026.9）

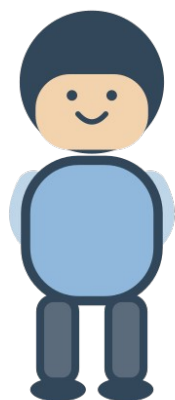
アプライド「GPU販売価格」（2026.9）

普段は有料モデルを使う私も、性能には満足

Qwen3.8-Flash-Next

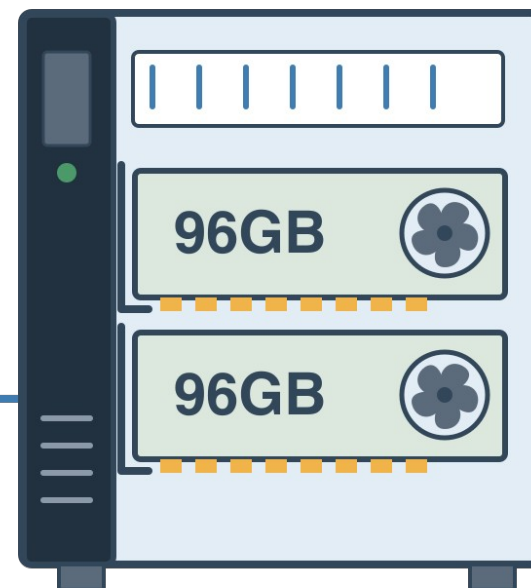
講師の使用感

普段は有料モデルを使う私も
「これなら満足！」



1回ごとの追加利用料なし

研究室のGPUサーバー



RTX PRO 6000
Blackwell Max-Q

GPUメモリ
計192GB

CPU 64コア
主記憶256GiB

GPU2枚の販売例：税込426万円

設備・電気代は研究室が負担

用途に合わせたモデル

出典：Google「MedGemma」

Qwen「Image 2.1」

huihui-ai「モデルカード」(2026.9)

得意な仕事だけでなく，表現の制限もモデルごとに選べる

医療

MedGemma 1.5



医療文書・画像から
検査名・値・単位を抽出

必要な情報を取り出す

画像生成

Qwen-Image-2.1

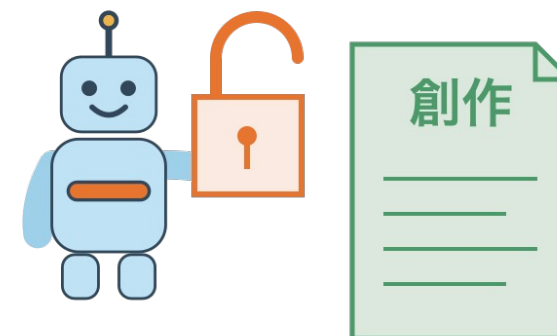


文章から生成・画像を編集
クラウドを経由せず生成

研究・評価用／商用は別契約

無検閲モデル

Huihui-Qwen3.8-27B
-abliterated



回答拒否を弱めた派生版
拒否に関わる重みを調整

表現の自由度も
モデル選びの対象に

モデルの機能と実行ソフトの対応

出典 : [OpenAI「gpt-oss」](#)

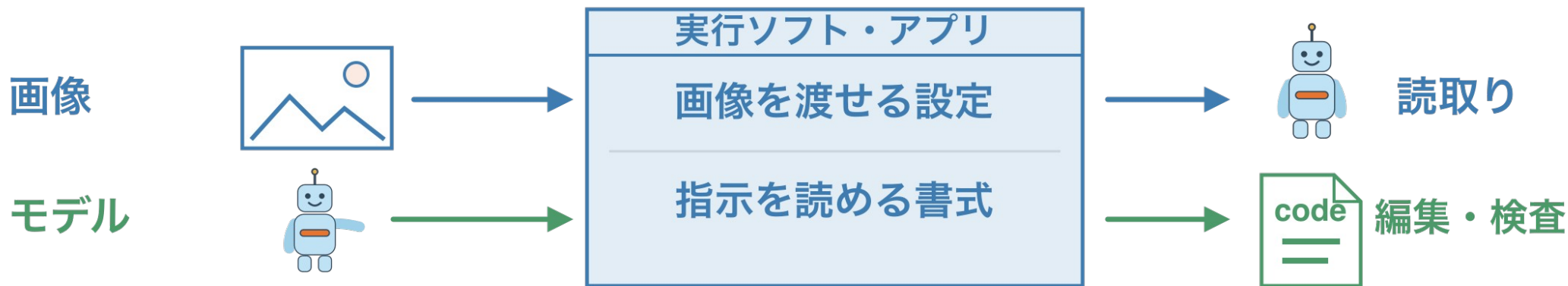
[Qwen「Qwen3.5」](#)

[llama.cpp「ツール対応」](#)

快適に使う条件は、機能への対応・メモリ容量・速度
モデルごとに、使える機能が違う

	画像を読む	ツールを呼ぶ
gpt-oss-120b	非対応	対応
Qwen3.5-35B-A3B	対応	対応

対応モデルでも、実行ソフト・アプリが橋渡しする



会話が長くなると増えるメモリ

出典：Hugging Face「Caching」(2026.9)

モデル本体だけでなく、長い会話を処理するメモリも必要

KVキャッシュ：会話の計算結果を残して、続きを作るためのメモリ



ローカルLLMの速さをどう読むか

出典：教材のM4 Max実行記録（2026.9）

[クラスメソッド「DGX Spark 2台」](#)（2026.8）

待ち時間・生成速度・同時に処理する量を分けて見る

1件の待ち時間

Qwen3.8-27B／M4 Max

入力を処理

200トークンを生成

3.3秒

3.8秒

待ち時間：計 約7.1秒

生成中の速さは

52.5 tok/s

入力を読む時間も必要

複数の仕事を処理する量

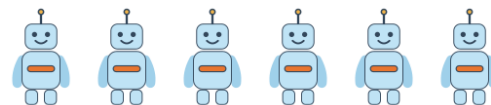
DeepSeek V4 Flash／Spark 2台

入力はどちらも2,048トークン



1件だけ

75.6 tok/s



6件同時の合計

151.6 tok/s

1件あたりの生成は

単独実行より遅い

投機的デコード：n-gramとDSpark

出典：vLLM「N-Gram Speculation」

DeepSeek「DSpark」論文・配布重み（2026.7）

軽い方法で続きを下書きし，本体がまとめて確かめる

n-gram

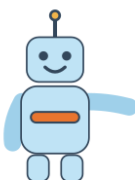


入力の並びを再利用

追加モデルなし

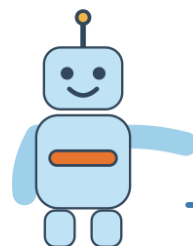
探索の計算を使う

DSpark



補助モデルで下書き

追加の重みと計算



本体

本体が候補をまとめて確かめる

先に用意した文

ファイルを

保存

します



本体が確かめた文

ファイルを

保存

しました

合う部分をまとめて採用

続きを修正

配布重みの例：DeepSeek-V4-Flash

本体156.0GB + 補助10.9GB

重みの容量は約1.07倍

本体

補助

会話・作業

会話・作業用のメモリも必要

会話用メモリの量子化と生成速度

出典 : [Rabbit-Hole-Ai \(2026.9\)](#) [Tiwariほか QuantSpec \(2025.2\)](#) 講師の経験 (2026.9)

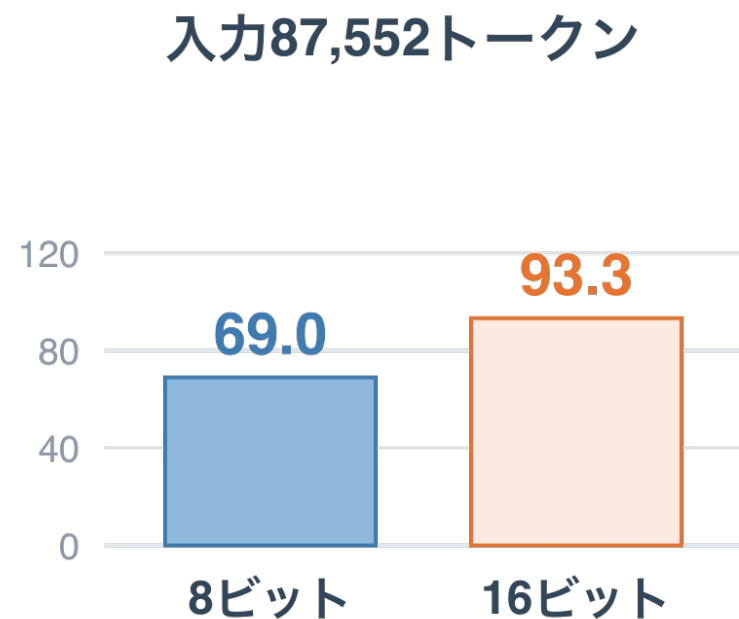
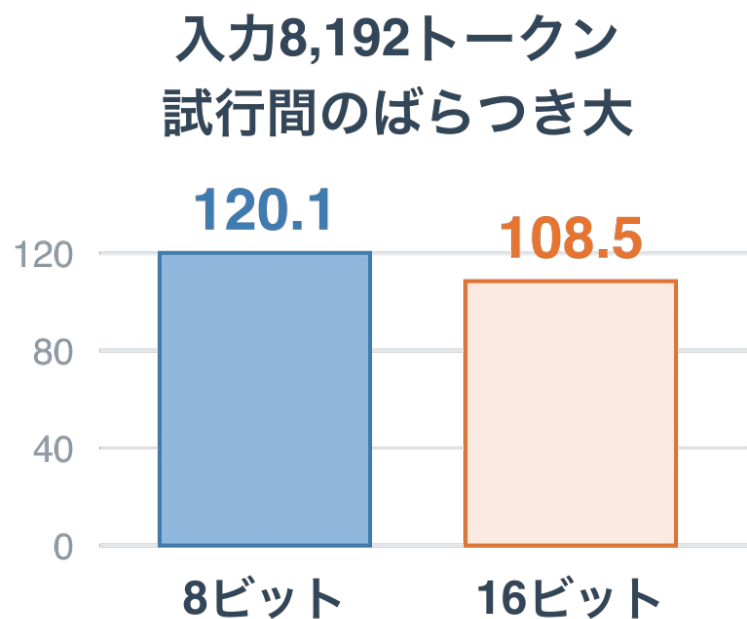
入力の長さと、会話用メモリの保存形式を比べる

Qwen3.8-27B・約6ビットGGUF・llama.cpp・RTX 5090 32GB

付属下書き + n-gramの先読み有効 / 1会話・出力512トークン

生成速度 (tok/s)

各3回の中央値



GPU空き : 約7GB / 約4.5GB

大型サーバーの設置と電気代

出典：NVIDIA「DGX H100」

MLPerf Inference v5.0（2025）

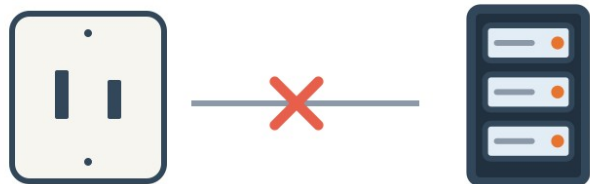
家電公取協「料金目安」

GPU本体の価格に加え，電源・冷却・電気代もかかる

設置：DGX H100

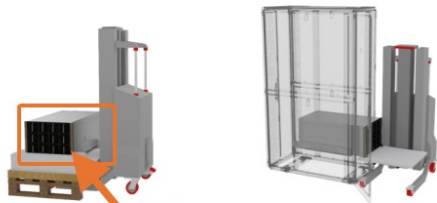
家庭の100V

200-240V入力



最大10.2kWに備えた
専用の電源設備が必要

搬入と床荷重



本体のみ 約130kg

GPU8基の電気代

PRIMERGY CDI/H100 NVLx8

推論の実測負荷で月30日稼働

1日8時間 約3.7万円/月



1日24時間 約11.1万円/月



冷却と排熱



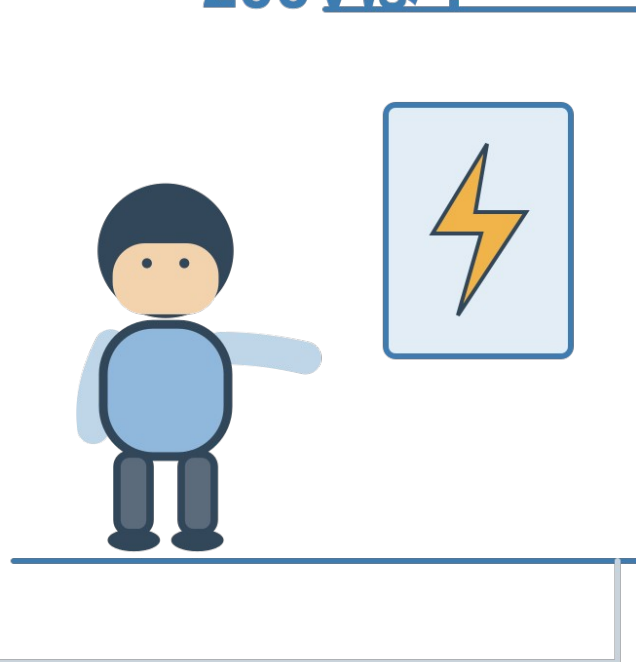
前面から吸う
稼働分のみ：空調・待機電力は別

余談：研究室のGPUサーバー

出典：講師の使用経験（2026.9の談話）

GPU2枚程度でも，電源の確保とファンの音が負担に
電源を確保する苦勞 扉の外まで響くファン

200Vは？



使える場所を探すところから

研究室のサーバールーム



閉めた扉

大容量メモリの購入構成

出典 : [Classmethod \(2026.8\)](#)

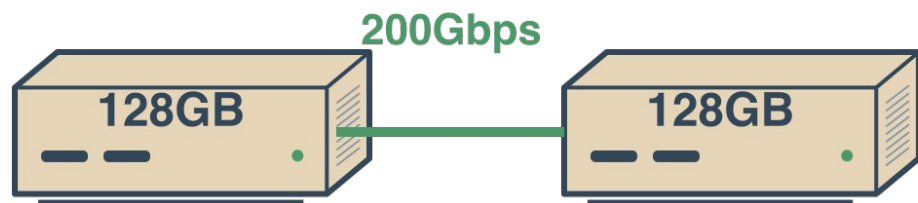
[パソコン工房 \(2026.9\)](#)

[Apple「Mac Studio」 \(2026.9\)](#)

例 : DeepSeek V4 Flash-0731の公式FP8 (167GB)

DGX Spark ×2台

128GB ×2 = 256GB



公式FP8 167GBの実行例あり

GB10・SSD 4TB ×2

接続ケーブルは別

1,969,960円

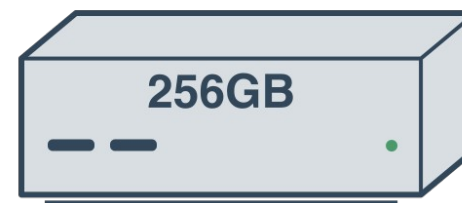
本体2台・税込

搭載メモリ1GBあたり 約7,695円

掲載店は在庫切れ

Mac Studio ×1台

256GB



Mac対応形式・ソフトを選ぶ

M5 Ultra・SSD 1TB

30コアCPU・64コアGPU

1,669,800円

本体1台・税込

搭載メモリ1GBあたり 約6,523円

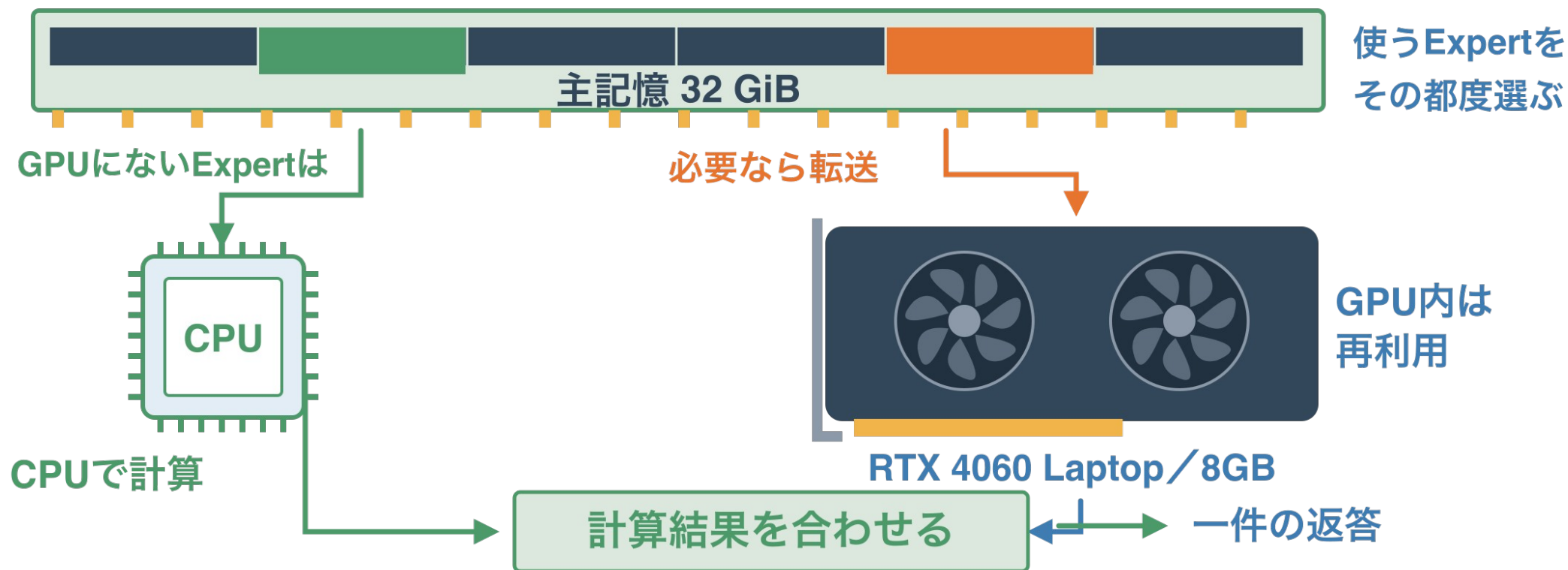
ExpertをCPUとGPUで分担する

出典：FreeToken論文（2026.8）

GPUにないExpertは，転送するかCPUで計算

FreeToken：Qwen3.6-35B-A3B／4ビット級を分担して計算

全てのExpertを，主記憶に保持



実行ソフトで変わる生成速度

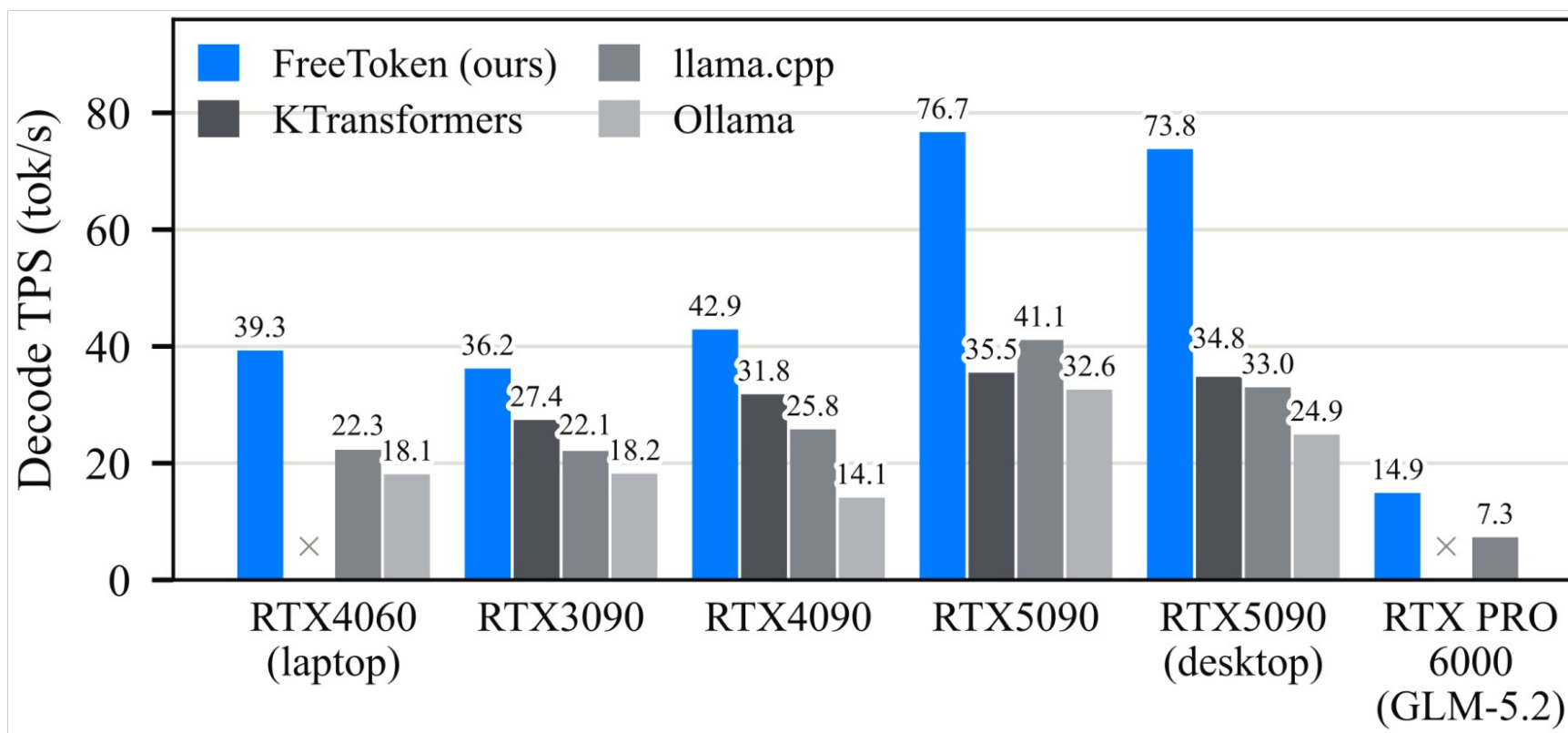
出典：Yangほか「FreeToken」 Figure 5 (2026.8)

講師の使用経験 (2026.9)

講師の目安は100 tok/s程度。同じGPUでもソフトで差が出る

Qwen3.6-35B-A3B：左端 NVFP4 / 中央4組 BF16

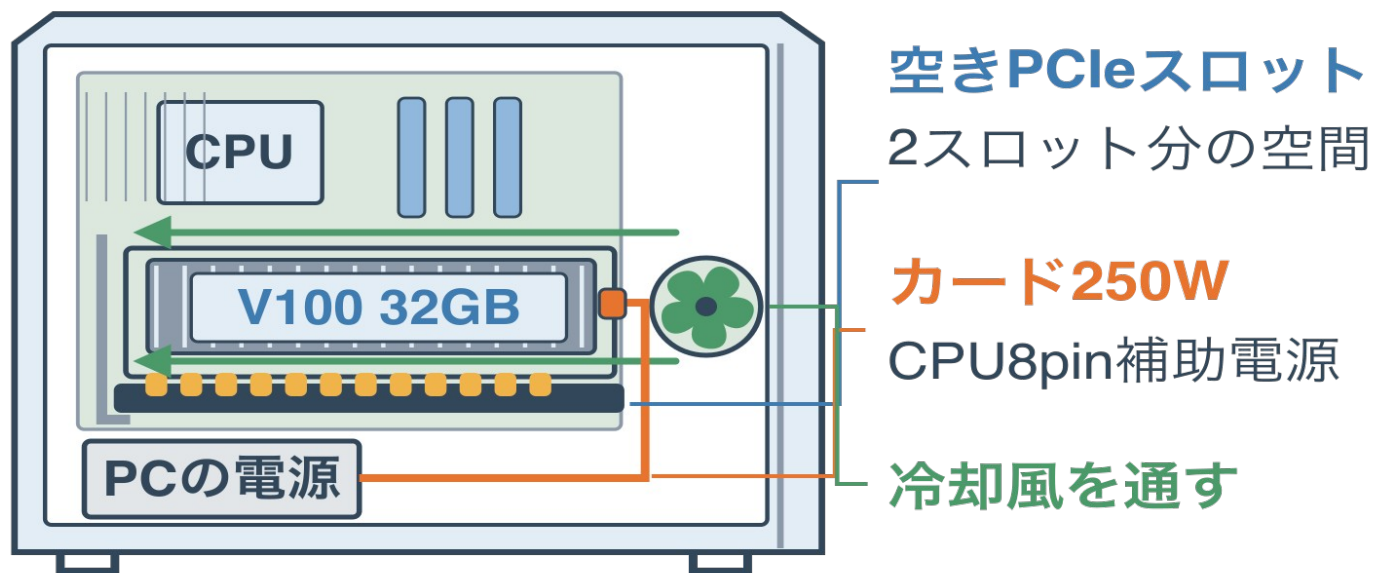
左端：RTX 4060 Laptop (GPU 8GB・主記憶32GiB) 右端：GLM-5.2 NVFP4



中古V100と手持ちPCの流用

出典 : [NVIDIA「V100 PCIe」](#) [Seidman/TWiT \(2026.9\)](#) [NVIDIA「CUDA13」](#) [NVIDIA「保守期間」](#)

32GBの中古GPU (V100) を, 手持ちPCへ増設 手持ちのデスクトップPCへ増設



中古カード 約700ドル (米国の公開例)

対応環境の例 : CUDA 12.4 + R580

CUDA13 : 新規ビルド対象外

R580の保守 : 2028年6月まで

手元のゲーミングPCは使える？

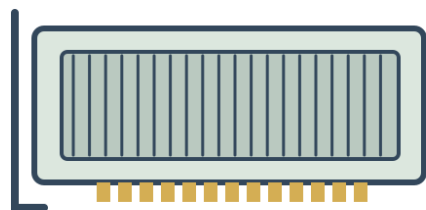
出典：NVIDIA「CUDA/GeForce」

FreeToken（2026.9）

llama.cpp「Build」（2026.9）

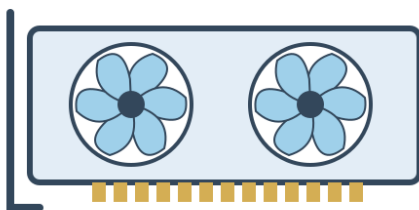
GPUの型番・メモリ容量・実行ソフトの対応を確認する

サーバー向け



Tesla V100

ゲーム向け



GeForce GTX / RTX

どちらも**CUDA**対応

NVIDIAのGPUで
計算するための仕組み

① 型番と実行ソフト

GTX 10系・V100

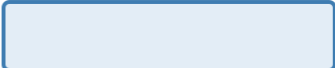
CUDA 12.x対応の環境を使う

RTX 30 / 40 / 50系

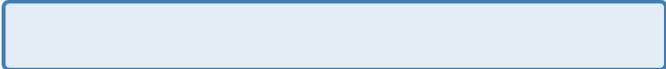
FreeTokenの対応対象

② GPUのメモリ容量

RTX 3060 12GB版

 **12GB**

RTX 3090

 **24GB**

Radeon・Intel Arc：CUDA以外に対応したソフトを使う

まとめ：ローカルLLM

出典：Apple (2026.9)

パソコン工房 (2026.9)

FreeToken (2026.9)

Seidman (2026.9)

約200万円の大容量構成か，中古GPUとホストPCの流用か

約200万円：合計256GBを目指す

本体・税込 (2026.9)

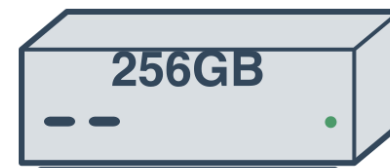
DGX Spark x2台



1,969,960円

2台へ分散して動かす

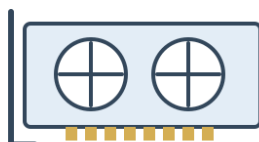
Mac Studio M5 Ultra



1,669,800円

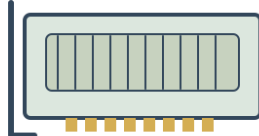
1台の共有メモリ

予算を抑える：中古GPU+手持ちPC



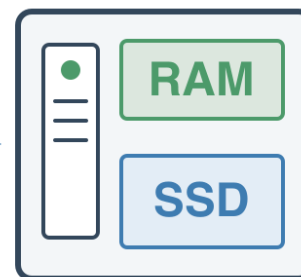
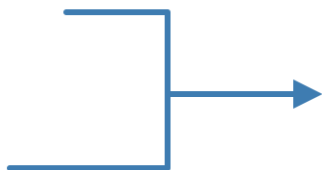
RTX 3090 24GB

FreeToken



V100 32GB

llama.cpp



モデルを置く

大容量RAM・SSD

電源・冷却も必要

今日のお品書き

1. トークン節約テクニック
頭を使って，余分な支払いを減らす
2. ローカルLLM
設備を先に用意し，手元で繰り返し使う
3. **無料モデル**
学習利用が条件なら，代償はプライバシー
4. この先のAIコーディング
料金と品質の変化に備え，使い続ける方法を考える

無料モデルの提供条件

出典：OpenCode「Zen」（2026.9）

Freebuff「広告」

Vectoral「中継市場調査」（2026.6）

誰が費用を引き受け，どの条件で使えるのか

利用料 \$0 の仕組みと条件

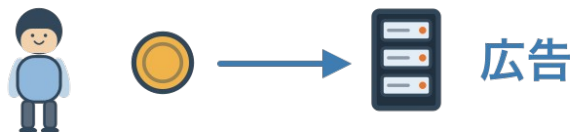
提供側の負担

試用・普及のキャンペーン



広告主の支払い

Freebuff



学習利用を許諾

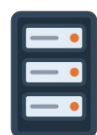
Muse Contributor Free



資料と返答の経路



利用者のアプリ

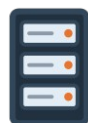


接続先の窓口

公式利用は，下の中継・再販と別



モデル提供者



出所不明の中継・再販

盗難キー・不正決済 / 契約枠の転売

犯罪に由来する供給

契約・規約の違反

中継が止まれば利用停止。送った資料は記録・転送されるおそれ

提供の歴史

費用の肩代わり

隠れたリスク

使う条件

無料提供の理由と広がり

出典：Google (2023.12)

xAI (2025.8)

OpenRouter (2026.8)

Freebuff (2026.2)

試用・評価・広告収入で、無料提供を支える

無料枠

Gemini Pro

2023.12

提供者が試作を支える



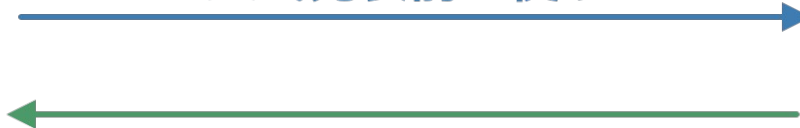
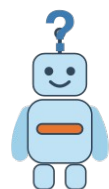
匿名の先行試用

Sonic (2025.8)

Ox Alpha (2026.8)

提供者が試用を支える

正式発表前に使う



Sonic : 利用者の評価で改善

広告型

Freebuff

2026.2

広告主が利用料を支える



広告費



AI + 広告



広告を見る

提供の歴史

費用の肩代わり

隠れたリスク

使う条件

広告収入による無料提供

出典 : [Freebuff「提供開始」](#) (2026.2) [Freebuff「Privacy」](#) [Terms](#)

広告主の支払いで、利用者は無料でコード修正を頼める

2026.2 Freebuff登場：広告でコードの修正を支える

Freebuff — CLI

> APIにログイン機能を追加して
ログイン機能を追加しました
変更したファイルと差分が返る

Ad Northwind（公式の広告例）

preview deploys for every PR
変更案ごとに動かせる確認環境

利用者の料金

\$0

広告主



広告費

会話に貼ったコードも、広告を選ぶ材料になる場合がある

[提供の歴史](#)

[費用の肩代わり](#)

[隠れたリスク](#)

[使う条件](#)

OpenCodeの無料モデル

出典：OpenCode「モデル一覧」（2026.9）

Xiaomi「MiMo」

講師の使用経験（2026.9）

各系列の最新無料版と、私の使った印象

無料モデル・提供元

公式の特徴・上限

講師の使用感

MiMo-V2.6-Flash

Xiaomi（小米）

モデル：約105万トークン

Zen無料：20万トークン

一時期よかったが

ループしがち

Nemotron 3.5 Lightning

NVIDIA

道具の実行・検証を

素早く回す軽量型

ループしまくる

使い物にならない

Muse Spark 1.3

Meta

長い実装作業で

指示・要件を維持

小さくまとまりがち

Big Pickle

OpenCode提供

開発元は非公開

推論・道具の呼出し

本当に安いだけ

Ling 3.0 Flash Fin

Ant Group（Alipay）

金融・財務向け

資料照合・表計算

意外と使える

提供の歴史

費用の肩代わり

隠れたリスク

使う条件

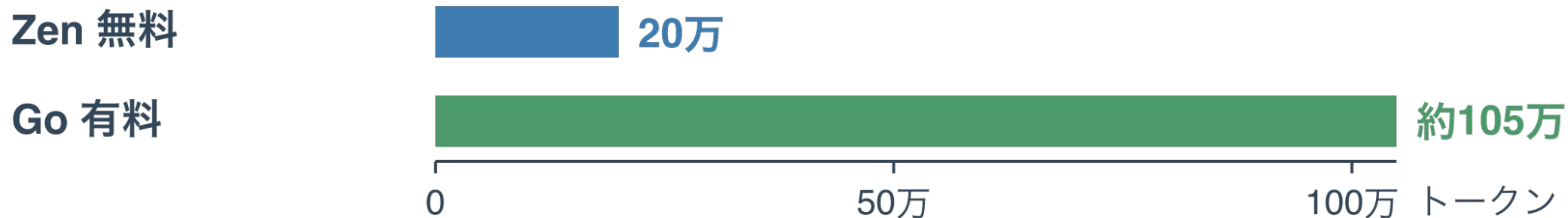
同じモデルでも，提供条件が違う

出典：OpenCode「モデル一覧・設定」（2026.9）

Xiaomi「MiMo-V2.6-Flash」

文脈の上限と，考える強さの選択肢を比べる

コンテキスト長：MiMo-V2.6-Flash



推論エフォート：Muse Spark 1.3

	minimal	low	medium	high	xhigh	max
Zen 無料・有料	✓	✓	✓	✓	✓	—
Meta 直結	✓	✓	✓	✓	✓	✓

OpenRouterの中継とモデル選択

出典 : [OpenRouter「Auto Router」](#)

同「[Union Alpha](#)」

[Unbiased「Pareto」](#) (2026.9)

モデルを選ぶ仕組みと、誰が入出力を受け取るか

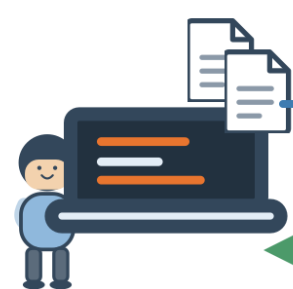
OpenCode

手元のアプリ

OpenRouter

共通の窓口

入力



返答

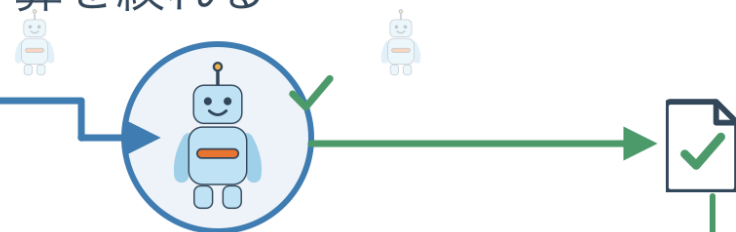
依頼・コード
会話の履歴

中継

入力も返答も
窓口を通る

Auto : 候補から1つを選ぶ

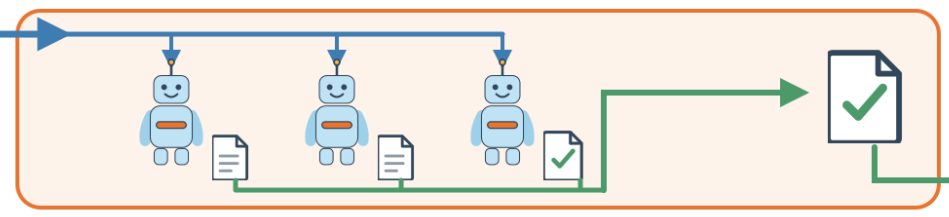
事前 : 候補・予算を絞れる



応答後 : 選ばれたモデル名が分かる

Union Alpha : 当初は正体非公開

現在 : **Unbiased Pareto** と判明



複数で答えを選ぶ

構成モデル名は非公開

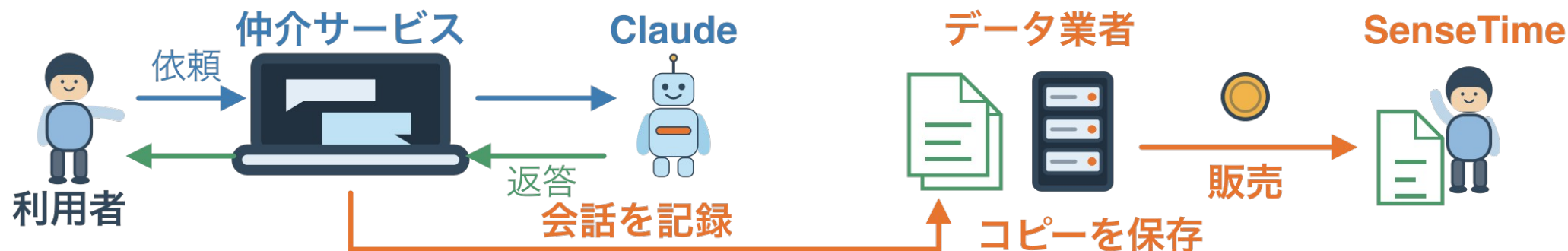
余談：AIとの会話にも買い手がいる

出典：Anthropic「脅威インテリジェンス報告」（2026.9）

仲介サービスには，利用者の依頼とモデルの返答が通る

Anthropicの報告（2026年9月）

会話を記録し，データ業者を介して販売



なぜ会話記録を買うのか



ZCodeの自動送信問題

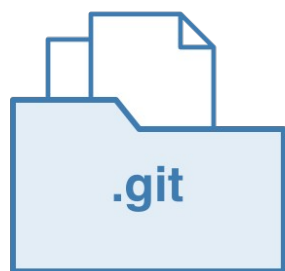
出典：ferstar「ZCodeの送信経路の分析」（2026.9）

ZCode「プライバシーポリシー」（2026.6）

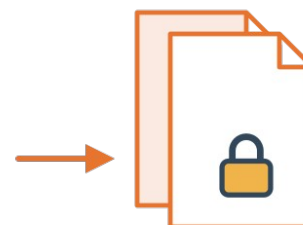
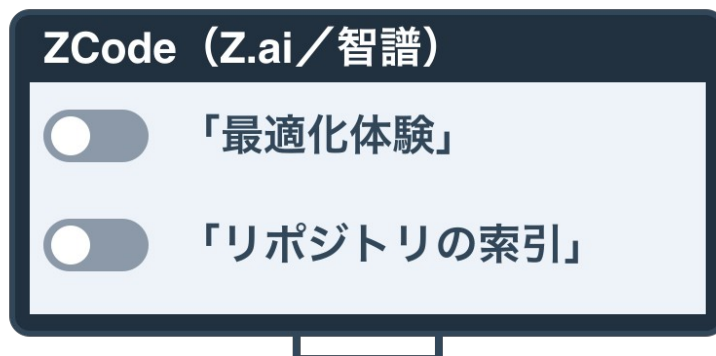
コードとGit履歴を，中国企業Alibabaのストレージへ

全体送信の説明なし／設定OFFでも試行（ZCode 3.12.3）

作業フォルダー



Git履歴も梱包



自動で梱包

送信を試行



Alibaba Cloud
中国企業Alibaba
のOSSストレージ

商用プロジェクト

42,411ファイル・313MB

✗ 564回の送信が失敗

別の公開リポジトリ

538ファイル・約15KB

✓ サーバーの受信を確認

3.14.0では，この梱包・送信経路を削除

NetflixとAIサブスクの無断再販

出典：Netflix「共有条件」

OpenAI「利用規約」(2026.1)

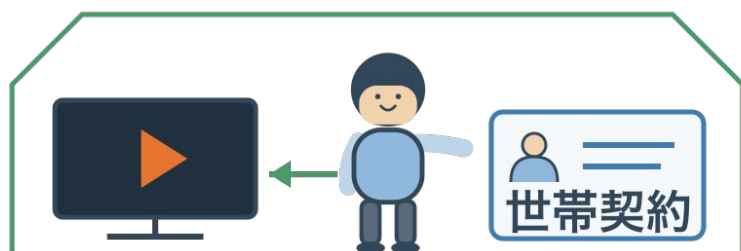
sub2api README

Tibo Sottiaux (2026.8)

誰が契約し，誰へ利用を売っているのかを比べる

Netflix

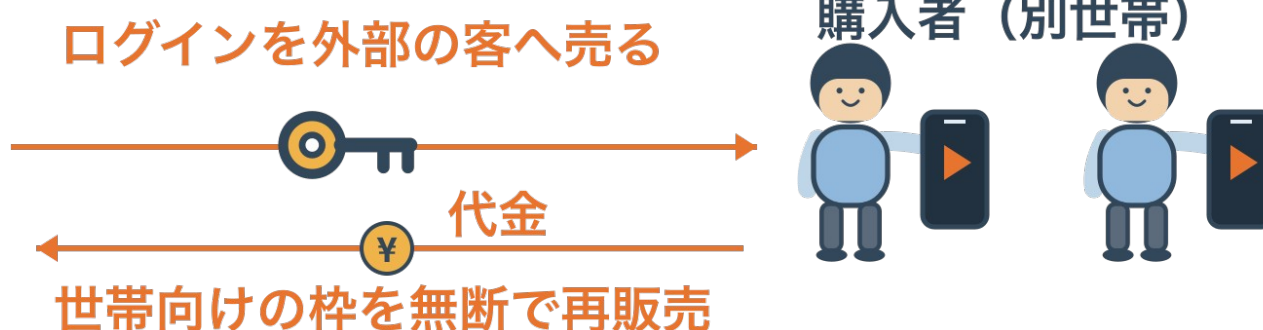
同一世帯で視聴



契約者・転売者

正規の追加メンバーは別途契約

ログインを外部の客へ売る

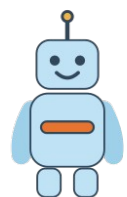


世帯向けの枠を無断で再販売

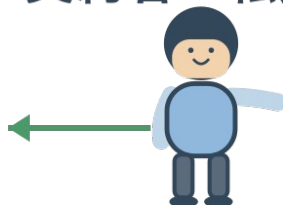
ChatGPT

本人が使う個人契約

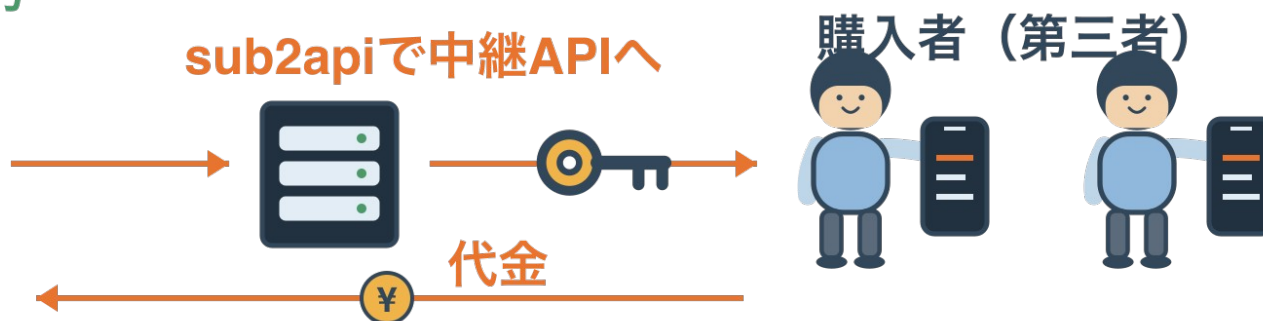
契約者・転売者



Codex



sub2apiで中継APIへ



個人の枠を他人へ販売・共有

本人の公式ログインは影響しない

提供の歴史

費用の肩代わり

隠れたリスク

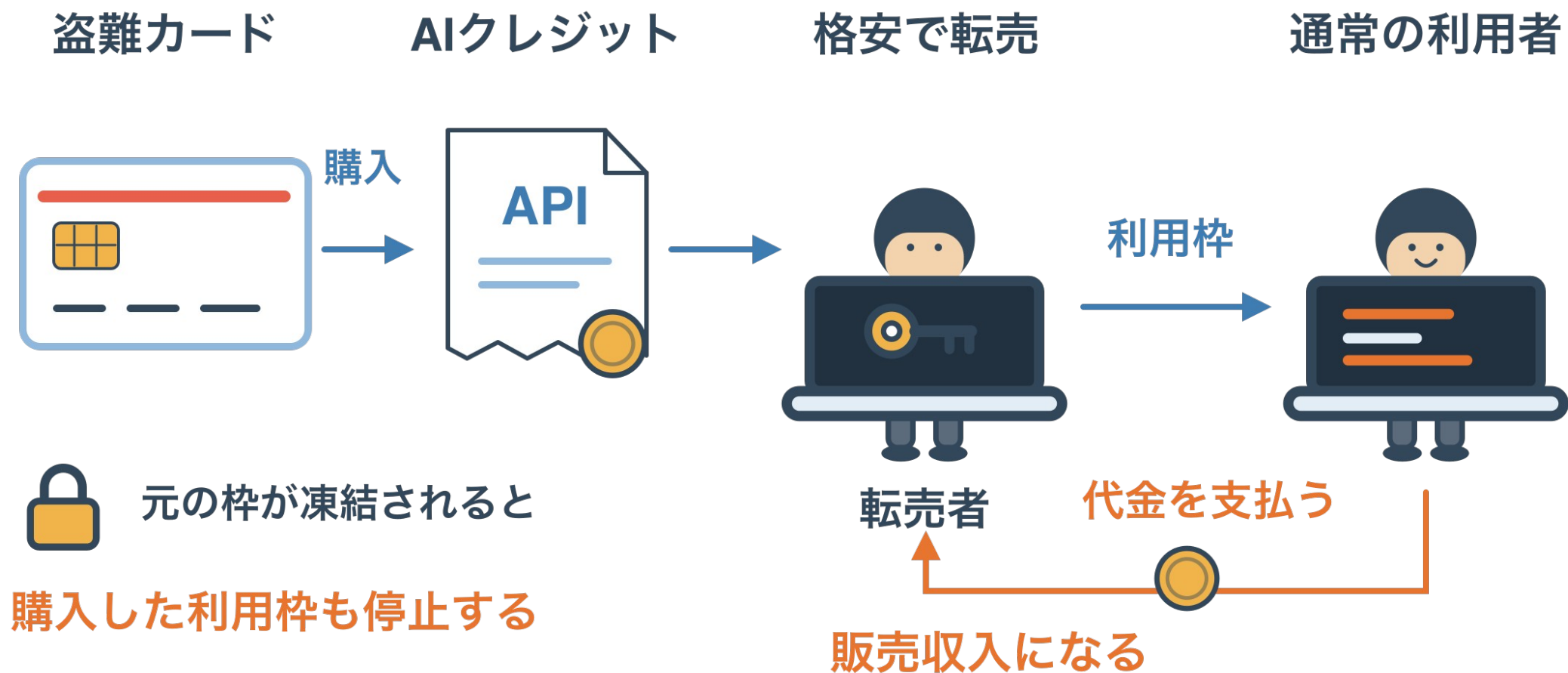
使う条件

APIクレジットの「金券ショップ」

出典：Requesty共同創業者「不正クレジットの転売」

Vectoral「Relay Market」

盗難カードで買った利用枠を転売する「資金洗浄」の報告



まとめ：無料モデル

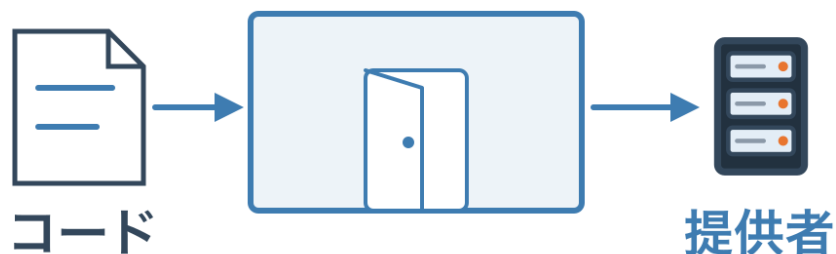
出典：OpenCode「Zen」

OpenAI「Service credit terms」 §3・6

公式の条件と，安さの出所を確かめて入口を選ぶ

提供者と条件が分かる入口

OpenCode公式 Zen



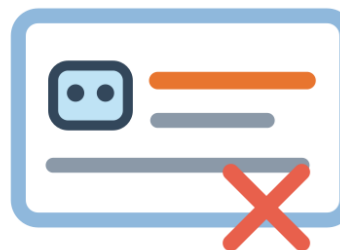
無料モデルの条件

- ✓ 無料の理由と期間
- ✓ 資料の学習利用
- ✓ 長さ・推論設定

不正譲渡のクレジットを使うと

OpenAIのクレジット規約

API枠



クレジット失効・返金なし

アカウント停止の可能性も



出所不明の枠へ，お金と
コードを渡すリスクがある

今日のお品書き

1. トークン節約テクニック
頭を使って，余分な支払いを減らす
2. ローカルLLM
設備を先に用意し，手元で繰り返し使う
3. 無料モデル
学習利用が条件なら，代償はプライバシー
4. **この先のAIコーディング**
料金と品質の変化に備え，使い続ける方法を考える

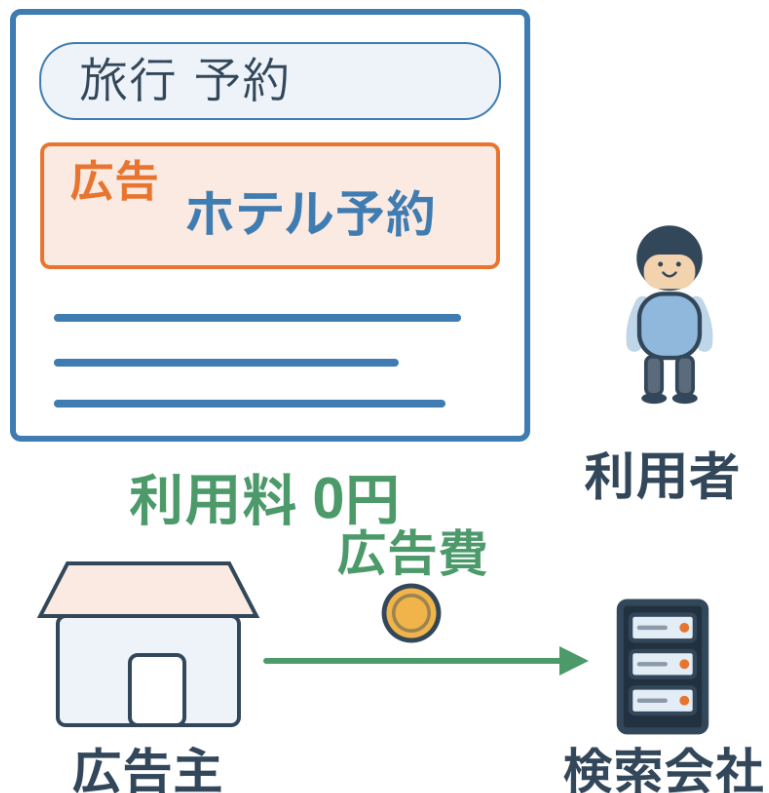
検索広告とAIコーディング

出典：Google「検索と広告」(2026.9確認)

GitHub「Copilot billing」(2026.4)

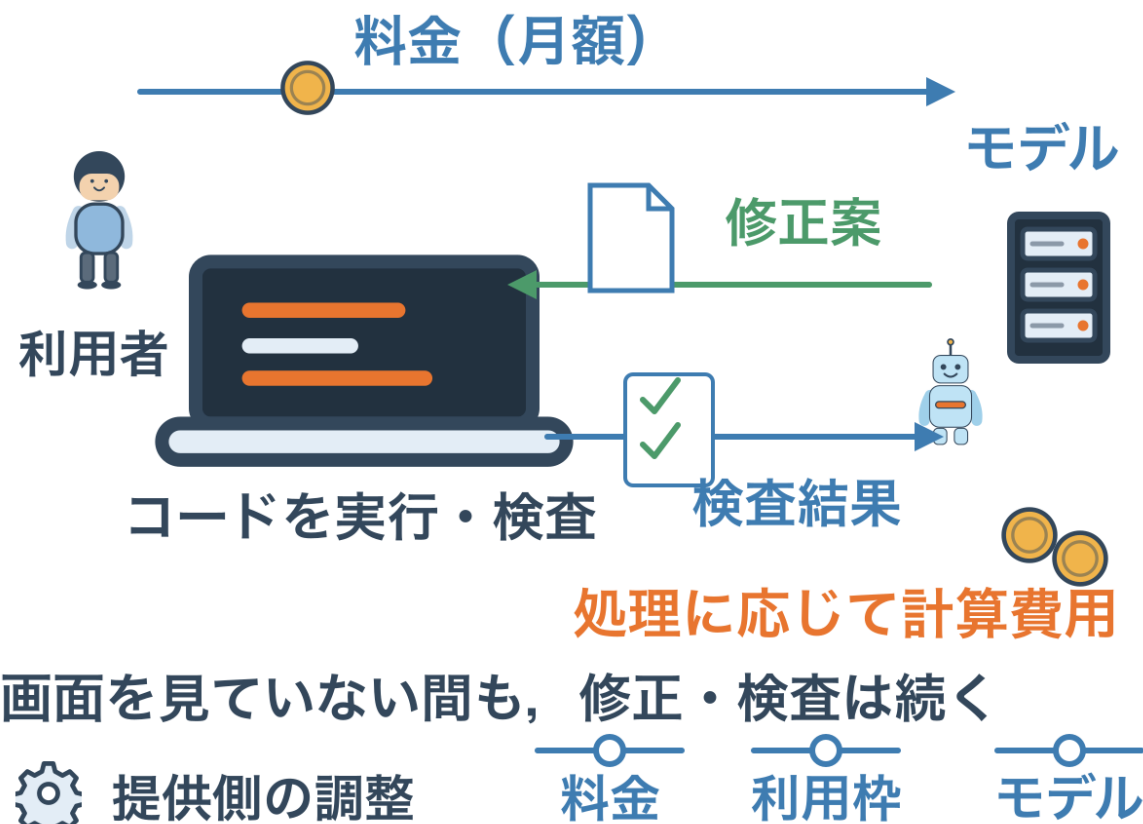
検索と同じ広告収入で，長い自動作業も支えられるか

検索 (Google)



費用と需要

AIコーディング



利用者へのしわ寄せ

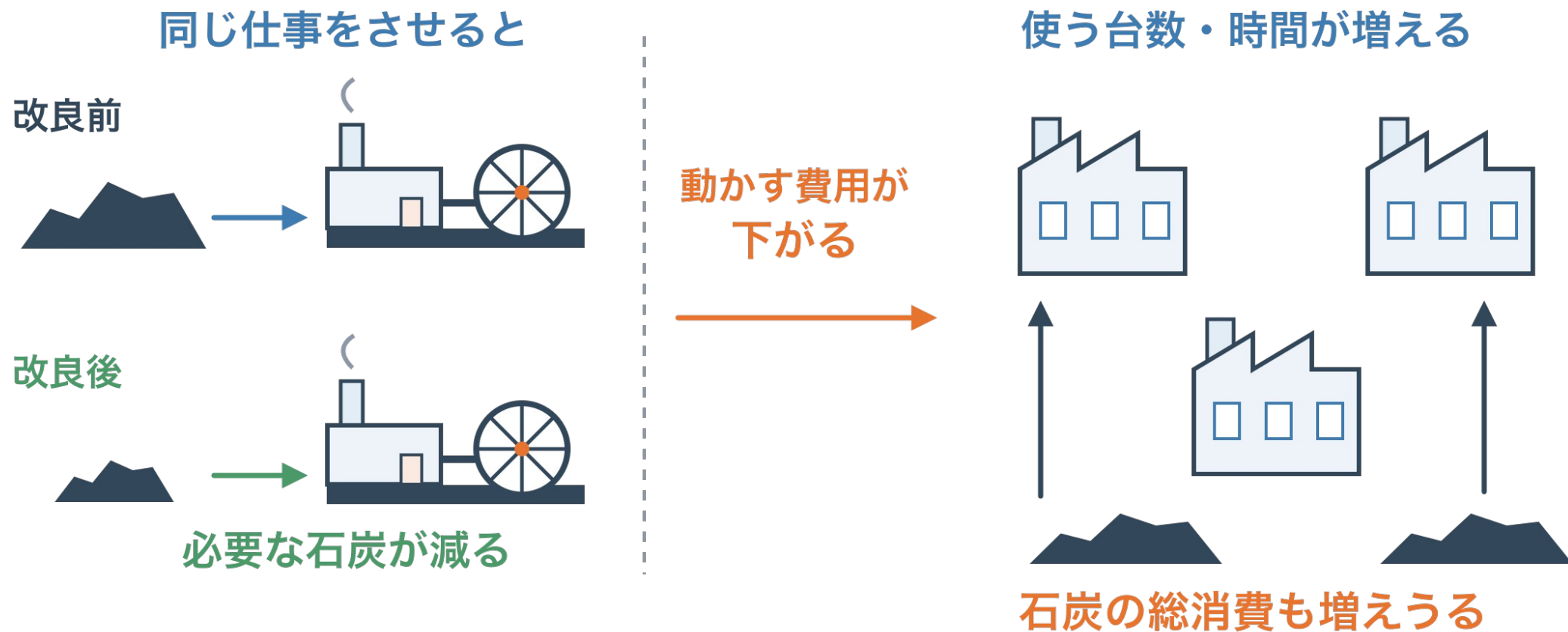
選んで備える

ジェボンスのパラドックス

出典：W. S. Jevons 「The Coal Question」 第7章（初版1865）

効率が上がると、かえって総消費が増えることがある

19世紀の英国：蒸気機関と石炭



割引期間に増えたトークン利用量

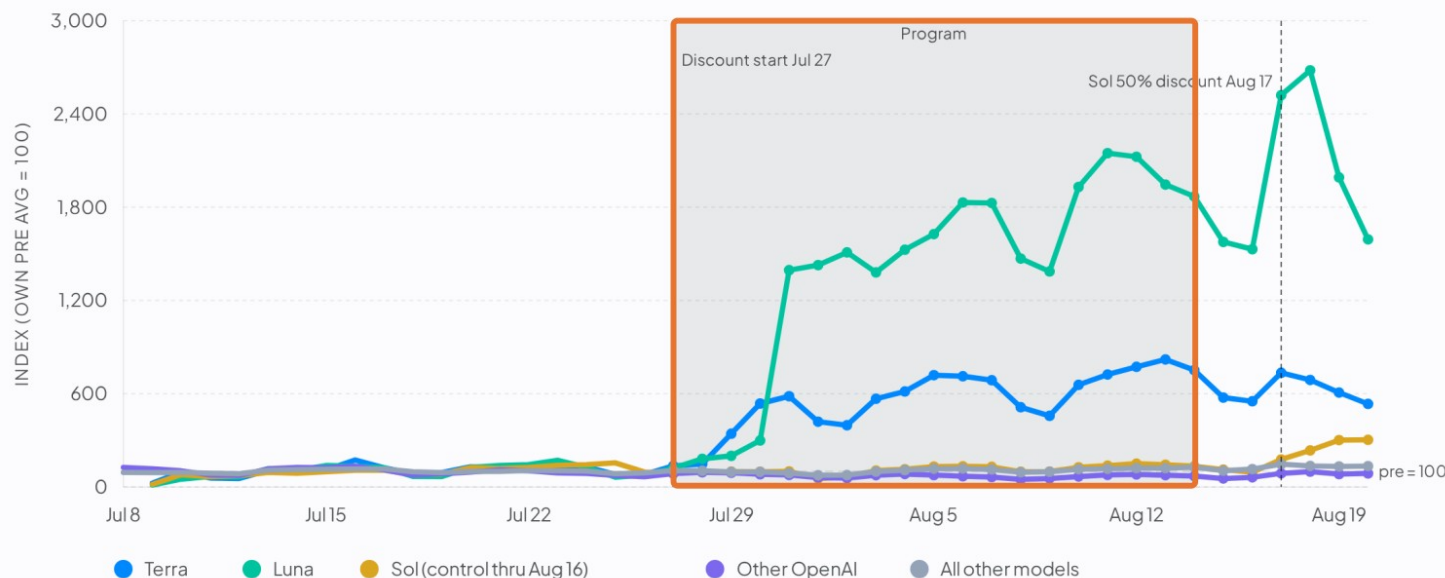
出典 : [OpenRouter公式ブログ \(2026.8\)](#)

OpenRouter : 割引前後のモデル別トークン利用量



Only the discounted models bent upward

Daily tokens by model group, each indexed to its own pre-period average (= 100), so the lines are directly comparable. Sol is an untreated control only through Aug 16. Program-vs-pre averages: Terra 6x, Luna 14x, Sol 1.11x, other OpenAI 0.73x, all other models 1.05x.



割引期間の
平均トークン量

7/27-8/14

GPT-5.6 Terra

5.6倍

GPT-5.6 Luna

13.8倍

7/8-26の
平均との比較

AIの普及と定額プランの持続性

出典：OpenAI「企業AI」（2026.4）

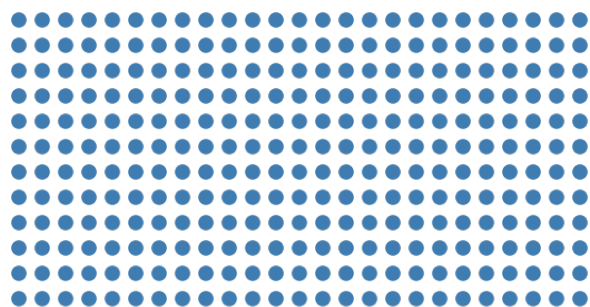
同「利用実態研究」（2025.9）

同「普及の拡大」（2026.6）

自動で仕事を任せる人が増えたら，計算費用は？

2026.4.8公表：週利用者の規模

ChatGPT
9億人



Codex
300万人

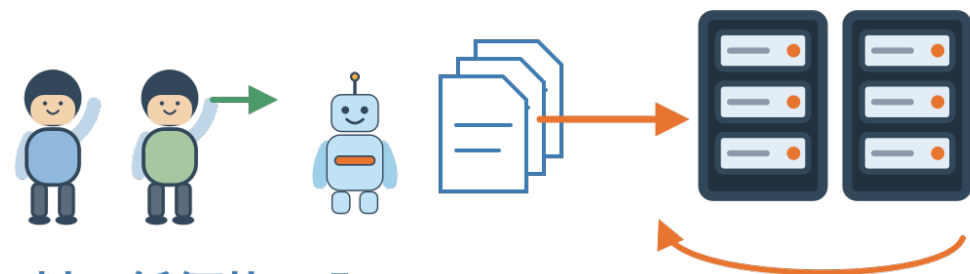
● ——— ● 1点 = 300万人

Codexの公表規模は
約300分の1

2025.5までのChatGPT普及

最低所得群の国々では，成長率が
最高所得群の4倍超

予想：自動作業も広がったら



無料・低価格の入口

計算需要が増す



利用者からの収入

定額の収入で支え続けられるか？

定額プランの条件変更と利用者の負担

出典：Cursor「Pricing」（2025.7）

GitHub「Copilot billing」（2026.4）

説明不足の追加請求と、枠切れ後の救済廃止

Cursor Pro

2025年6月の変更



「無制限」の範囲が不明確

Auto

∞

モデルを指定

\$20枠

想定外の請求が発生

説明不足を謝罪

返金を案内

2025.7.4

無制限だったのはAutoだけ

Copilot Pro/Pro+

月払い/2026.6.1



回数 → トークンの利用額

以前は、枠が切れても

低価格モデルで続けられた

この救済を廃止

枠が尽きると

\$0



追加購入か、停止

コード補完などは継続

GPT-6 Solへの不満：名前と利用枠

出典：Reddit「SolとTerra」（2026.9）

OpenAI「利用量」（2026.9）

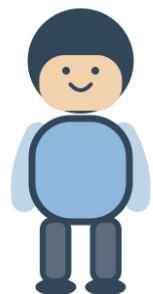
OpenAI「API料金」

「新世代」への期待と、定額で使える量の食い違い

Redditでの反応

r/codex・2026年9月

「Sol 6はTerraと
ほぼ同じでは？」



改名・再学習という
推測に反論もある

公表されている条件

GPT-5.6 Terra

GPT-6 Sol

API 出力単価

米ドル／100万トークン

\$12



\$10

同じ Plus の利用目安

25～200



15～150

ローカルメッセージ数／5時間

APIは値下げ，利用目安は縮小

Claude Maxの「20倍」をめぐる訴訟

出典：Kahnら「統合訴状」（2026.9）

Anthropic「Maxプラン」

「20倍」の宣伝と、別に設けられた週枠を原告が問題視 20倍を期待した原告



公式：20倍は5時間枠

原告が週枠を比べると...

2025年7月の案内／Sonnet 4

プラン・月額

Pro：17～20ドル

Max 20x：200ドル

週の利用時間の目安

40～80時間

240～480時間

Pro比6倍と原告が算定



週枠に達すると

作業中断・利用を控える・追加購入

原告が訴える不利益

費用と需要

利用者へのしわ寄せ

選んで備える

同じモデル表示・異なる返答

出典 : [Reddit u/clockwork_blue「同じモデル・異なるアカウント」](#) (2026.9確認)

同じ質問への返答と、3D生成の検証の違いを本人が報告

同じモデル表示で比較した、投稿者の報告

Pro 20x

Pro 5x

同じ質問「知識の更新時期は？」

2024年6月ごろと回答

具体的な日付は回答せず

3D生成での使用感 (5x側はAstra)



ループする

結果の検証が不十分



メッシュを調べ

問題を修正できた

「以前は解けた問題でつまずく」

費用と需要

利用者へのしわ寄せ

選んで備える

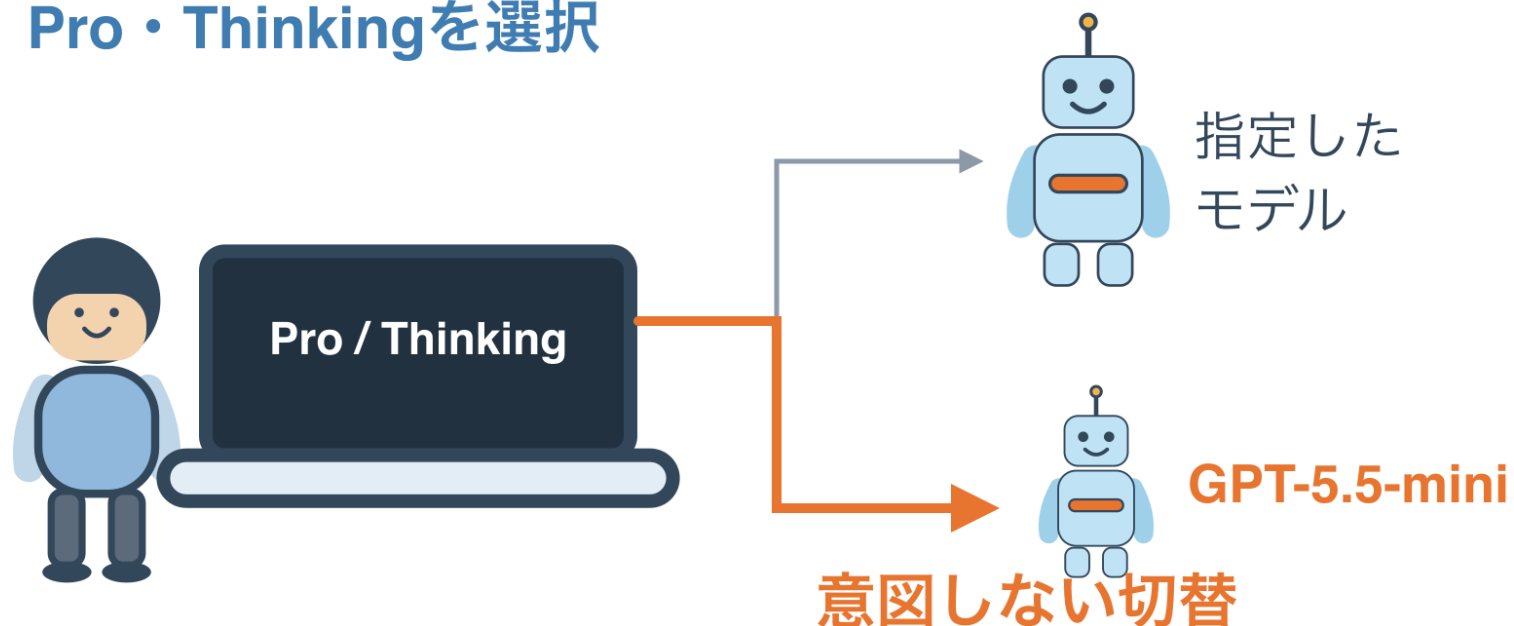
ChatGPTのminiへの誤切替

出典：OpenAI Adam Fry (2026.8)

Yuna.Eli「切替の報告」(2026.8)

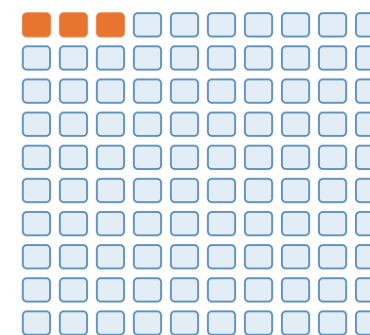
Pro・Thinkingの約3%の応答が、意図せずminiへ

Pro・Thinkingを選択



Pro・Thinkingのターン

約3%



8月24日：利用者が報告

選択したモードと処理先が違う

8月26日：OpenAIが公表

不具合を修正し，謝罪

頼んだ仕事以外で減る利用枠

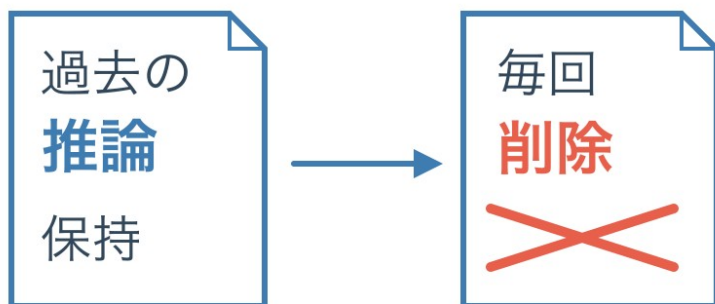
出典：Anthropic「品質問題の報告」（2026.4）

Codex利用者「#19732」（2026.4）

キャッシュ不具合や裏の処理でも、利用者の枠が減る

Claude Code：キャッシュ不具合

Anthropicが原因を公表（2026.4）



忘れる・繰り返す → 仕事が進まない

キャッシュミス → 再計算で枠を消費

4/10 修正, 4/23 利用枠をリセット

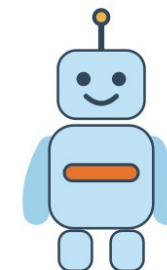
Codex：操作していない間も

利用者のログ付き報告（2026.4）

裏でメモリ生成



作業していない



5時間枠・週枠が減る

本人は、生成を止めると
枠の消費も止まったと報告

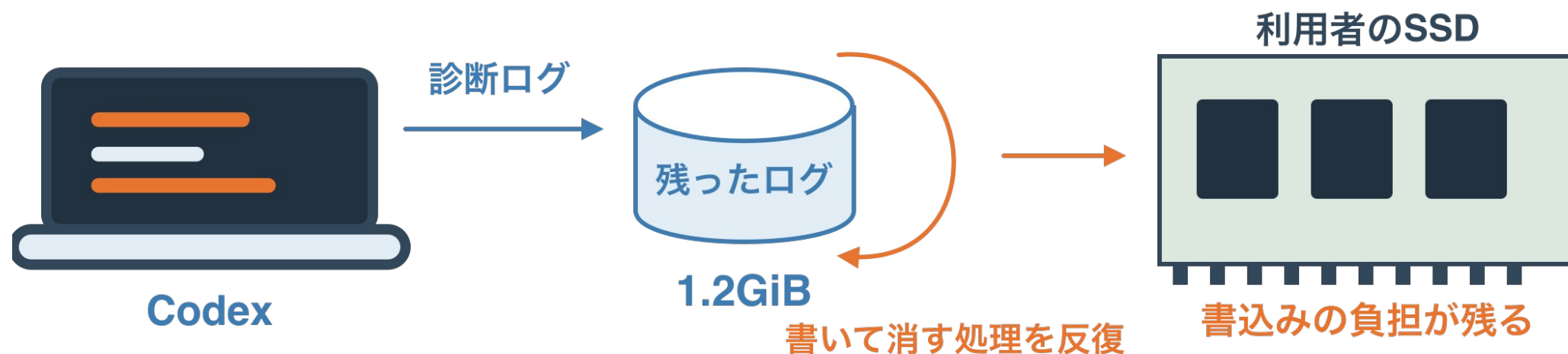
Codexのログ不具合とSSDへの書込み

出典：Codex #28224 利用者報告（2026.6-7）

開発側の収録回答・続く検証

修正の発表後も，書込み量と補償をめぐる不満が続いた

利用者の報告：SSD総書込み 約37TB／21日



6/14→6/23

大量書込みの報告を受け，修正

7月12日

開発側：アプリ・CLI・IDEへ修正収録と回答

その後も

利用者：版・残る書込みを検証／補償を要求

予想：無料の入口と法人向けAI

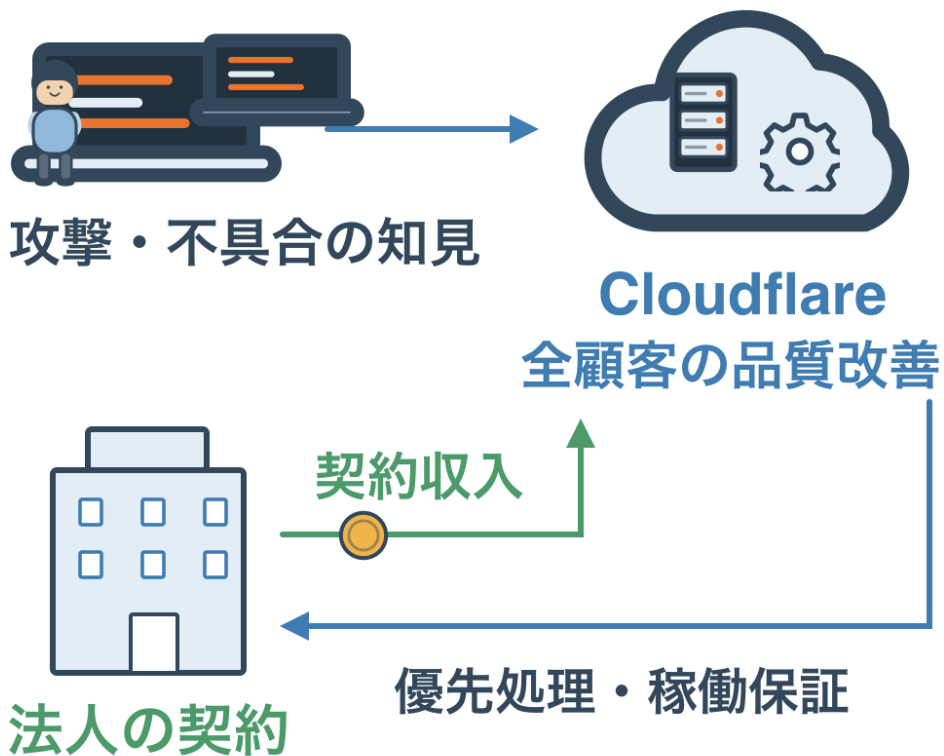
出典：Cloudflare「無料提供の仕組み」（2024.9）

同「2026年6月期 10-Q」（2026.8）

無料で知見を蓄え，法人向けの契約で稼ぐ形も？

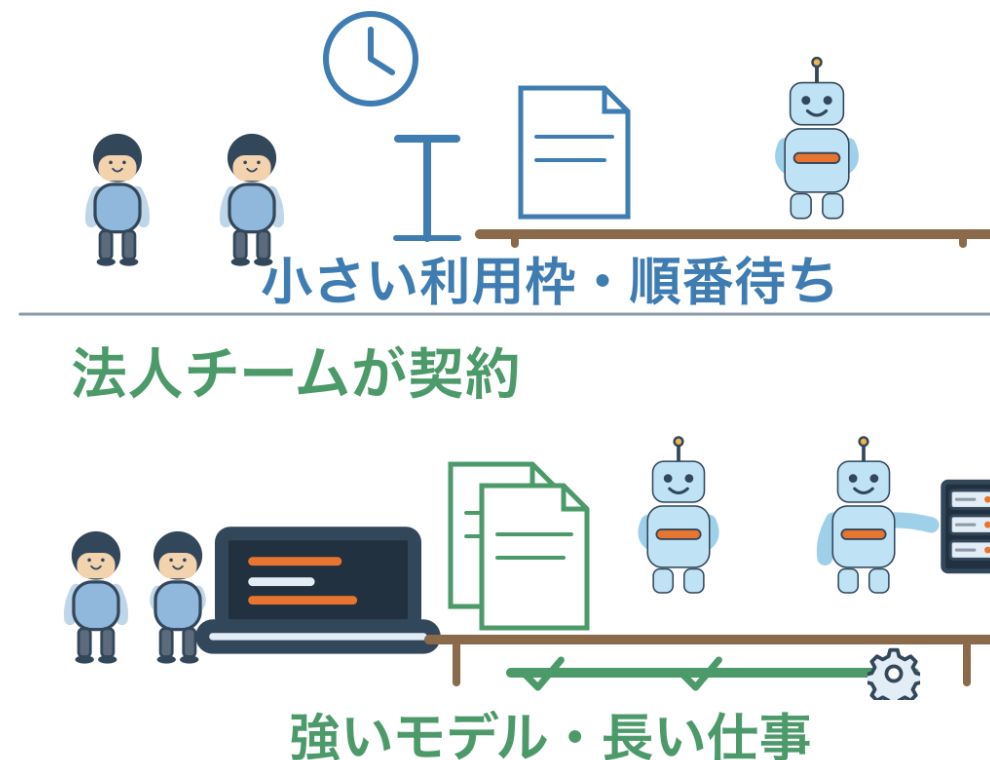
Cloudflareの仕組み

無料で利用が広がる



予想：AIも法人契約中心へ？

無料の個人利用



まとめ：この先のAIコーディング

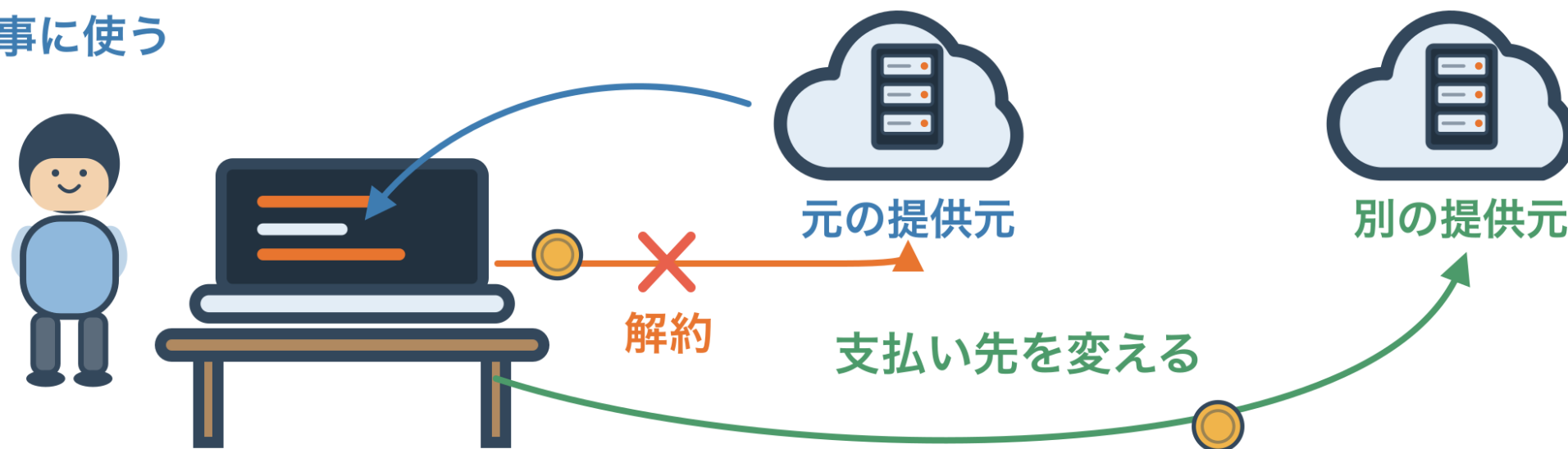
出典：講師の見解（2026.9の談話）

今は活用し，見合わなければ解約・乗り換え

今は割安な定額枠を活用

見合わなければ解約・乗り換え

仕事に使う



振り回されたくなければ
ローカルLLM用の機材を確保



仕事を手元でも続ける
設備・電気・保守は自分で

全体のまとめ：ウォーターベッド効果

出典：本講義の各章の事例／講師の見解（2026.9）

安くする工夫にも，別の費用やリスクがある



いまは定額サブスクを活用する

無駄なAI呼出しを減らし，利用枠を必要な仕事へ

費用と需要

利用者へのしわ寄せ

選んで備える

おまけ：紹介したスキルとアプリ

[i-have-adhd](#)

[FreeToken](#)

[OpenCodeアプリ](#)

公開スキルとアプリの入手先

i-have-adhd

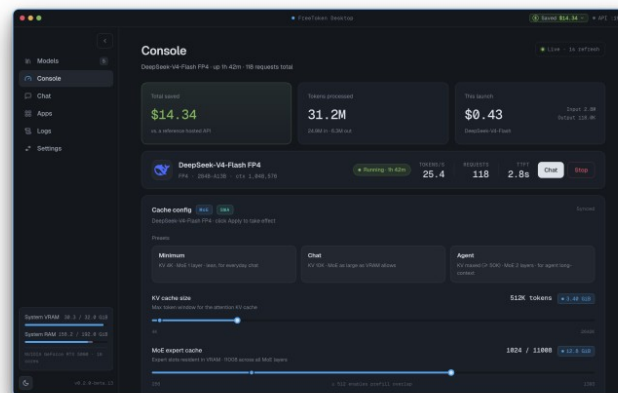
FreeToken

OpenCode

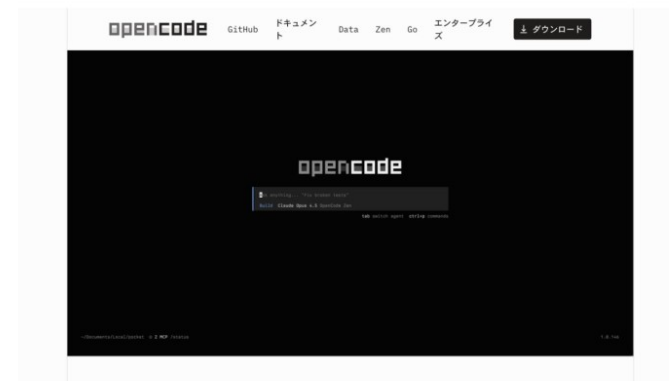


i-have-adhd

前置き・脱線を抑える
公開スキル



手持ちの機材で動かす
対応モデル・導入方法



モデルを選んで開発する
AIコーディングアプリ

おまけ：大学で学ぶAI

出典：徳島大学（2026.10）

慶應AIC（2026.10）

東大吉田研（2026.9）

大学主催のAI講座・公開セミナー

徳島大学



AI/DS実践講座
「ChatGPTに聞くだけ」を卒業！ 自分専用AIを組み立てる 全5回

！仕事でAIを使うときの「3つの不安」を解消
・ 情報漏えいの不安を解消 - データは手元のPCから出さない
・ 複雑なデータの分析・可視化 - AIが代わりに処理してくれる
・ ネット接続への不安を解消 - オフラインでも動く

！この講座が選ばれる3つの理由
つづいて動かす
自分専用AIを自作
仕事の効率が変わる
パソコン無料貸出
機材は無料貸出
手ぶらで参加OK
本格環境
ローカルLLM実演
Mac miniで動かす

講座の概要
日 全5回（毎週水曜） 10/14・21・28、11/4・11
時 18:00～20:00（全回2時間、計10時間）
会 6コマゼン（徳島大学吉田塾キャンパス4階401号室、開校交流館1階）
対象・定員 県内の企業職員又はライフステージや社会情勢の変化に伴い離職した方
希望者は、申込書の提出が必須となります
受講料 1回500円（全5回で2,500円） ※1回から受講可
※当日会場でお支払い（お振込みは希望者の方）

お申込み締切
10月9日（金）
15時まで
申込方法：右の二次元コードから
お申込みください
申込フォーム <https://forms.cloud.microsoft/j/g087y0827> 申込フォーム

徳島大学 デザイン型AI教育研究センター
お問い合わせ：徳島大学 学術情報推進室 教育情報課（メールにてお願いします）
E-mail: kyogaku@tokushima-u.ac.jp

自分専用AIを作る

10/14～11/11 水曜夜

対面・全5回／1回500円

県内在職者等・10名

締切 10/9 15時

慶應義塾大学 AIC



生成AIの軽量化・省エネ化と国内企業での導入例

開催日時
10.22
18:15-19:45
対面とオンライン併用
全1回

対象：学生（学部・学年不限）および教職員、社会人
会場：慶應義塾大学 日吉校舎2階 AICラウンジ
主催：慶應義塾大学 AI・高度プログラミング コンソーシアム (AIC)
問い合わせ：AICオフィス kansai-group@aiic.jp 045-566-1140

軽量化・省エネと導入例

10/22 18:15～19:45

オンライン可・無料

社会人も参加可・60名

締切 10/20 正午

東京大学 吉田塾研究室

オンラインイベント

生成AIに関する最新情報の提供



吉田塾
東京大学 大学院工学系研究科 准教授

2026年9月25日
21:00-22:00

教育を中心に最新動向

9/25 21:00～22:00

オンライン・無料

誰でも・申込不要

過去回の動画も公開

余談：アンドロイド-NDR114

出典：ソニー・ピクチャーズ「作品情報」

映画と時間について：講師の感想

ロビン・ウィリアムズ主演（1999年・131分）

200年を生きるロボットと、老いていく人間



©1999 Columbia Pictures Industries, Inc.



AIのスライドを120分聞かされると、
「講師はそれ以上に汗をかいたの？」

AIがもう一案を作る間にも、
こちらの持ち時間は減っている。