



Tech  
News  
Daily

# 2026.05.11

MORNING DISPATCH / Vibe Coder Bootcamp Tech News

## 今朝のホットな話題

1. 📰 **Codex CLI 0.130.0** — ``codex remote-control`` でスマホ/外出先から自宅マシンを操作可能に
2. 📰 **Claude Code 「Routines」** — ``/schedule`` で定期実行 AI ジョブを 1 行設定
3. 📰 **16人エンジニアチームが Anthropic 解約**  
→ **Codex+Cursor** へトークン効率議論が再燃



**5** トピックを整理。

 TEKION Group

# Codex CLI 0.130.0 – `codex remote-control` でスマホ/外出先から自宅マシンを操作可能に

## 🔍 何が起きた？

OpenAI が Codex CLI 0.130.0 をリリース。新しい `codex remote-control` コマンドにより、ヘッドレスかつリモート制御可能なアプリサーバーを起動できる。自宅のMacなどで動いている Codex に、外出先のブラウザ（事実上・スマホ）から接続して、作業継続や追加指示が可能になった。

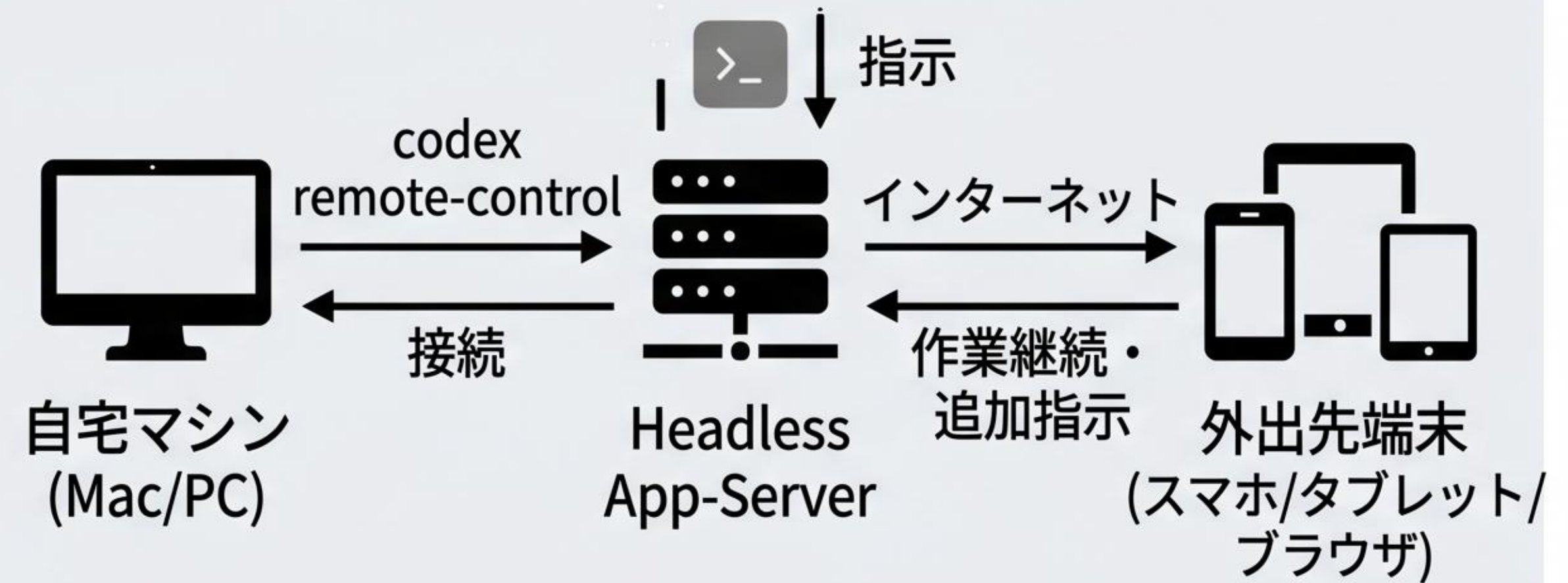
## 📌 主な変更点

- codex remote-control で headless app-server が起動、別端末から接続可能
- プラグイン共有が link metadata + discoverability controls 付きで強化
- Bedrock 認証が AWS console-login profile 経由でサポート

## 💡 なぜ重要？

外出先からでも強力な自宅マシン上のAI環境を利用できる。モバイルデバスによる開発フローの柔軟性が向上。

## 💻 `codex remote-control` の仕組み



## 👥 Community Response (数字・引用)

- @Codestudiopjkb: 日本語解説、368L
- @hAru\_mAki\_ch: 194L
- @Codex\_Changelog: 1,048L で告知
- @neromeron1014: tailscale + 自宅 MacbookPro で `codex\_mobile` を自作



## 🔍 何が起きた？

Claude Code に「Routines」機能が追加。ユーザーは簡単な「/schedule」コマンド1行で定期実行 AI ジョブを設定可能。cron 風に Claude がローカルまたはリモート環境で自走。

- ローカルとリモートの両方をサポート

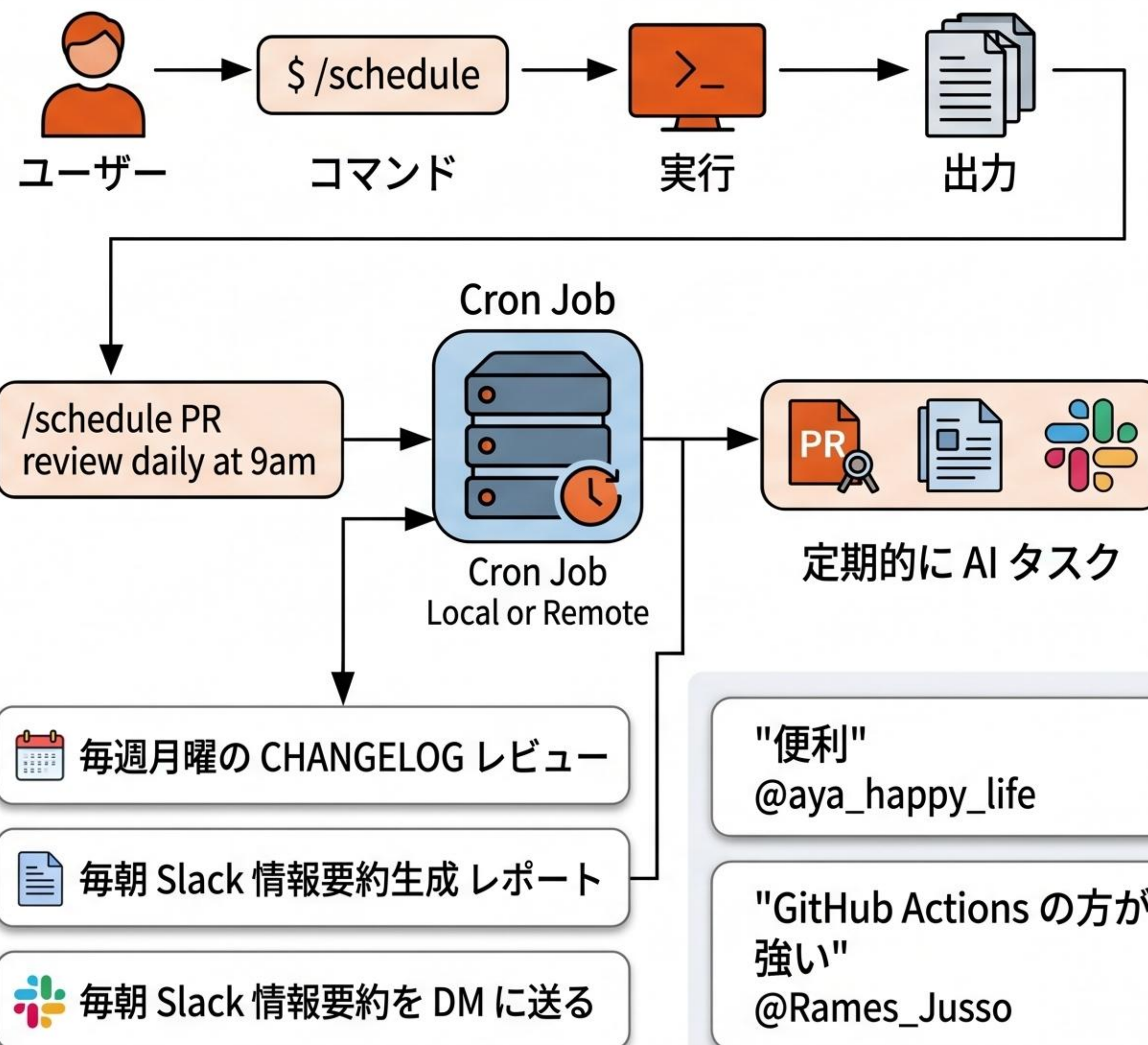
## 📌 主な変更点

- /schedule <prompt> コマンド1行で定期実行をセット可能
- ローカル/リモート両環境に対応
- PR レビュー、CHANGELOG レビュー、毎朝 Slack 情報要約送信などのユースケース

## 💡 なぜ重要？

- ワークフロー自動化による効率性の向上
- Claude Web 側の Routine (@zento\_ai の実装した「Slack 情報を DM に送る」など) に対応する CLI 版
- ユーザー評価は割れる：便利 (@aya\_happy\_life) vs. GitHub Actions が今は強い (@Rames\_Jusso)

Source: Anthropic公式



# 16人エンジニアチームが Anthropic 解約 → Codex+Cursor へ — トークン効率議論が再燃

## 🔍 何が起きた？

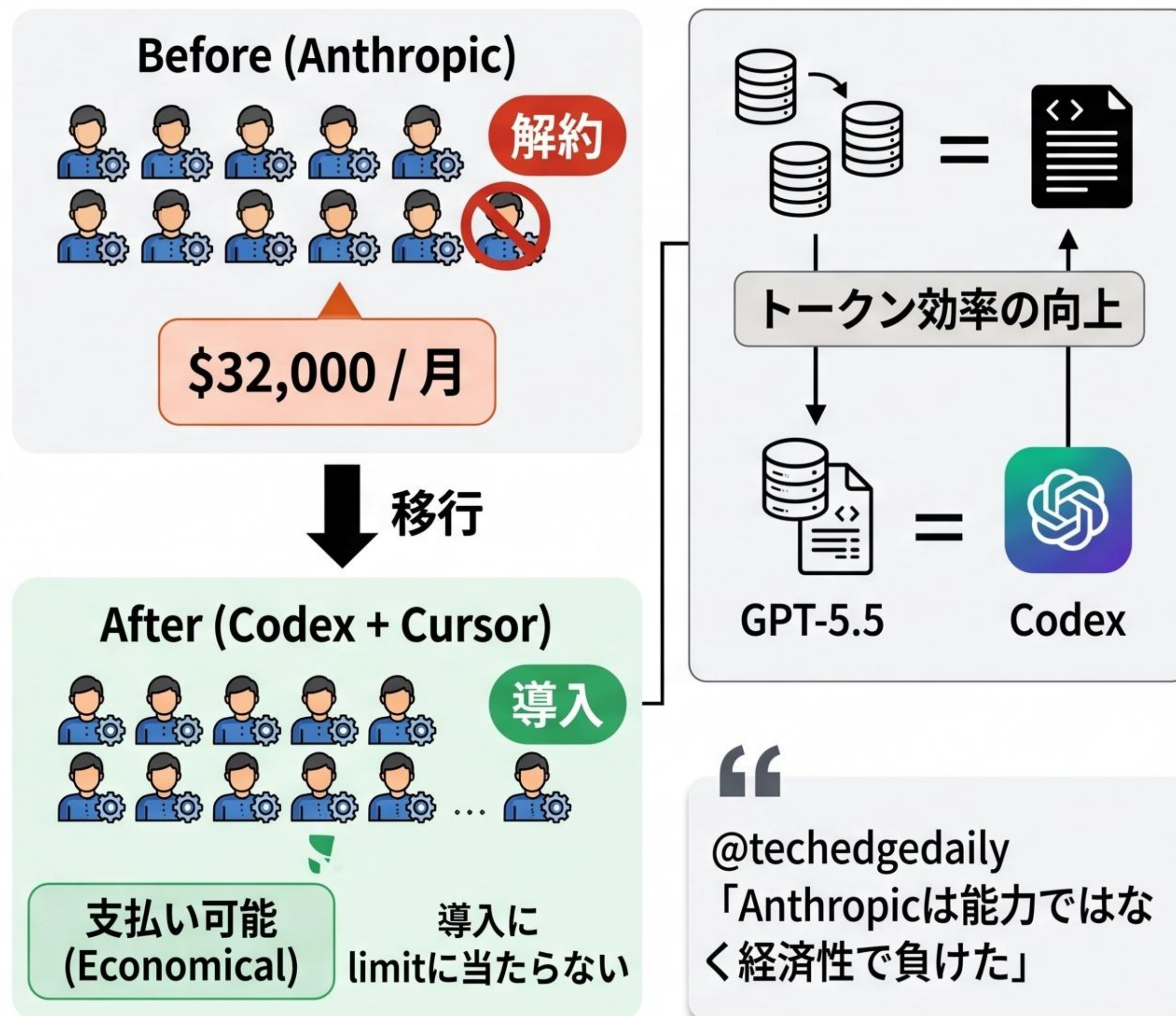
Morgan Linton率る16名のエンジニアチームは、月\$32,000に達するコスト（\$2,000/人）を理由にAnthropicを正式に解約。高性能でトークン効率が向上したCodex + Cursor（GPT-5.5ベース）中心の運用に全面的に切り替えた。

## 📌 主な変更点

- 16人 × \$2,000/月 = \$32,000/月のコスト構造を支払い切れず。
- CodexはGPT-5.5で精度・トークン効率が大幅に向上、「so token efficient」。
- Cursor Composer 2もチーム運用でlimitに当たらない。
- Droidも並行検証中、レビューは引き続き人手で多くこなし、最終判断は人間前提を維持。

## 💡 なぜ重要？

本事例は、生成AI活用において「能力」よりも「経済性」が決定打となることを示す。特に大規模チームでは、1人当たりのコストとトークン効率が全体の運用性を左右する極めて重要な要素となる。



# OneBiz: Claude Code でプレゼン資料を 5 ステップで作る — PowerPoint/Slides を捨て HTML+GitHub Pages へ



Anthropic via X

## 🔦 要点

OneBiz が X long-form で「Claude Code を使ったプレゼン資料の作り方 5 ステップ」を公開。PowerPoint/Google Slides は AI 時代の制約になっていると主張。

## 🔧 具体的な手法 / 使いどころ

- 5ステップ: スクリプト準備 / Claudeプロジェクト + システムプロンプト / チャット編集 2 回まで / Claude Code 細部 / GitHub Pages
- 「構造変更はチャット、細部は Claude Code」の使い分け
- チャット 3 回目以降はコンテキスト汚染で品質劣化

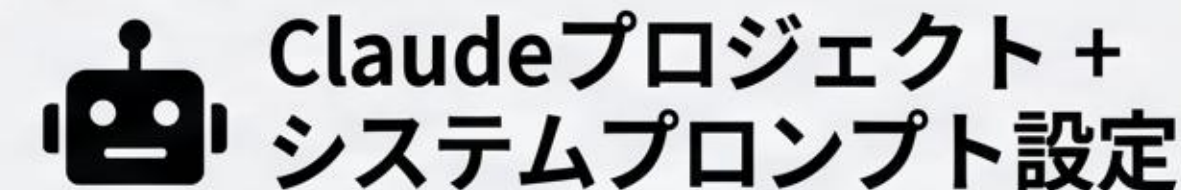
## 🌱 なぜ刺さるか / 学び

- AI との効率的な対話による自動プレゼン制作
- 従来のツールの限界を突破し、モダンなHTMLプレゼンへ
- Claude プロジェクト + システムプロンプトテンプレ → チャット編集 → Claude Code仕上げ → GitHub Pagesデプロイのワークフロー

### Step 1



### Step 2



### Step 3



### Step 4



Claude Codeで細部調整

### Step 5



GitHub Pages  
へデプロイ

**5** 5 ステップ  
の制作ワークフロー



チャット編集は  
**最大 2 回**  
まで

⚠️ 3回目以降は品質劣化

- ### 💡 システムプロンプト推奨要素
- 単一 HTML、CSS&JS インライン
  - 各スライド最大 3 行
  - ダークBG (actual dark colors like #0a0a0a applied) + アクセントカラー
  - CSS Animation & 矢印キー操作

# Chormex に GPT-Realtime-2 でリアルタイム翻訳 + Codex で同時要約 — 「ブラウザ = リアルタイム AI OS」

## 🔍 何が起きた？

CHOI が自身の Chrome 拡張「Chormex」に OpenAI の GPT-Realtime-2 と Codex を統合。

Chrome 内のあらゆる音声（YouTube, ライブ配信, 会議, プレゼン）をリアルタイムで翻訳・要約・ノート化が可能に。

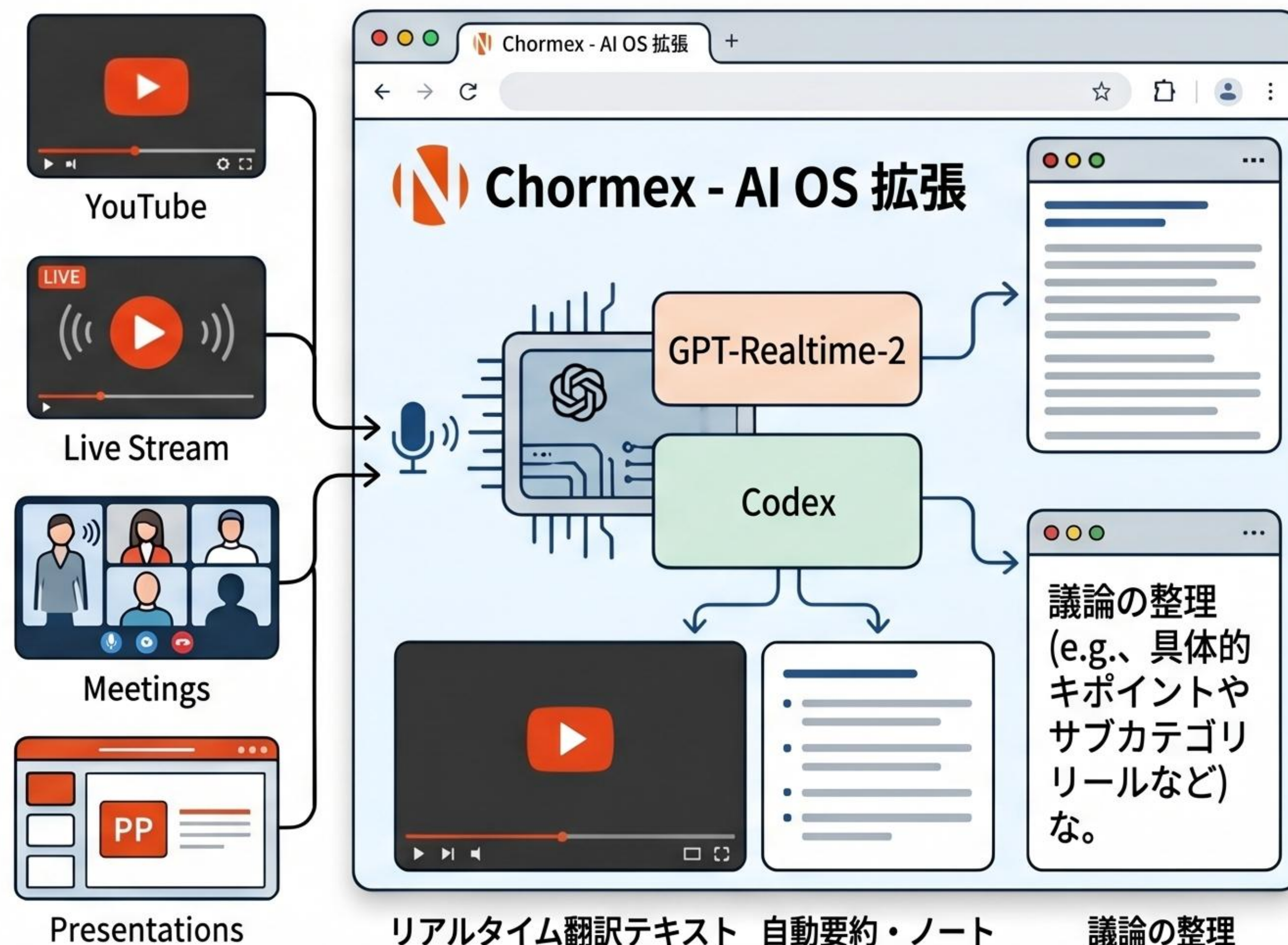
## 📌 主な変更点

- GPT-Realtime-2 による低レイテンシ翻訳 + LLM 解析の同時実行
- 対象は全 Chrome 音声：YouTube, ライブ配信, 会議, プレゼン
- 操作例："Summarize this." / "Turn this into notes." / "Organize the discussion."
- GPT-Realtime-2 商用ユースケース 2件目（OpenAI に続く組み込み事例）

## 💡 なぜ重要？

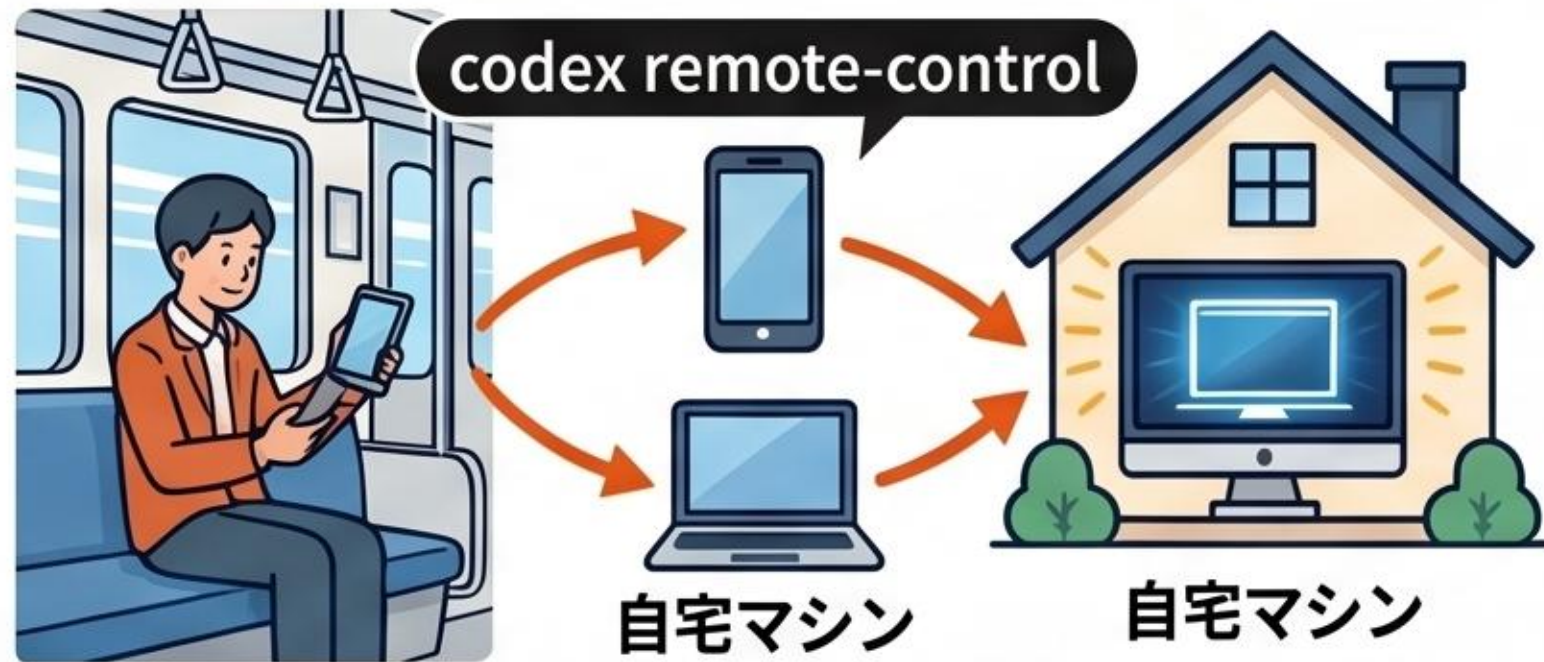
- ブラウザが単なる閲覧ツールから、リアルタイム AI オペレーティングシステム（AI OS）へ進化。
- 様員の音声・動画コンテンツを瞬時に多言語翻訳・構造化し、生産性を飛躍的に向上。

## ブラウザ = リアルタイム AI OS

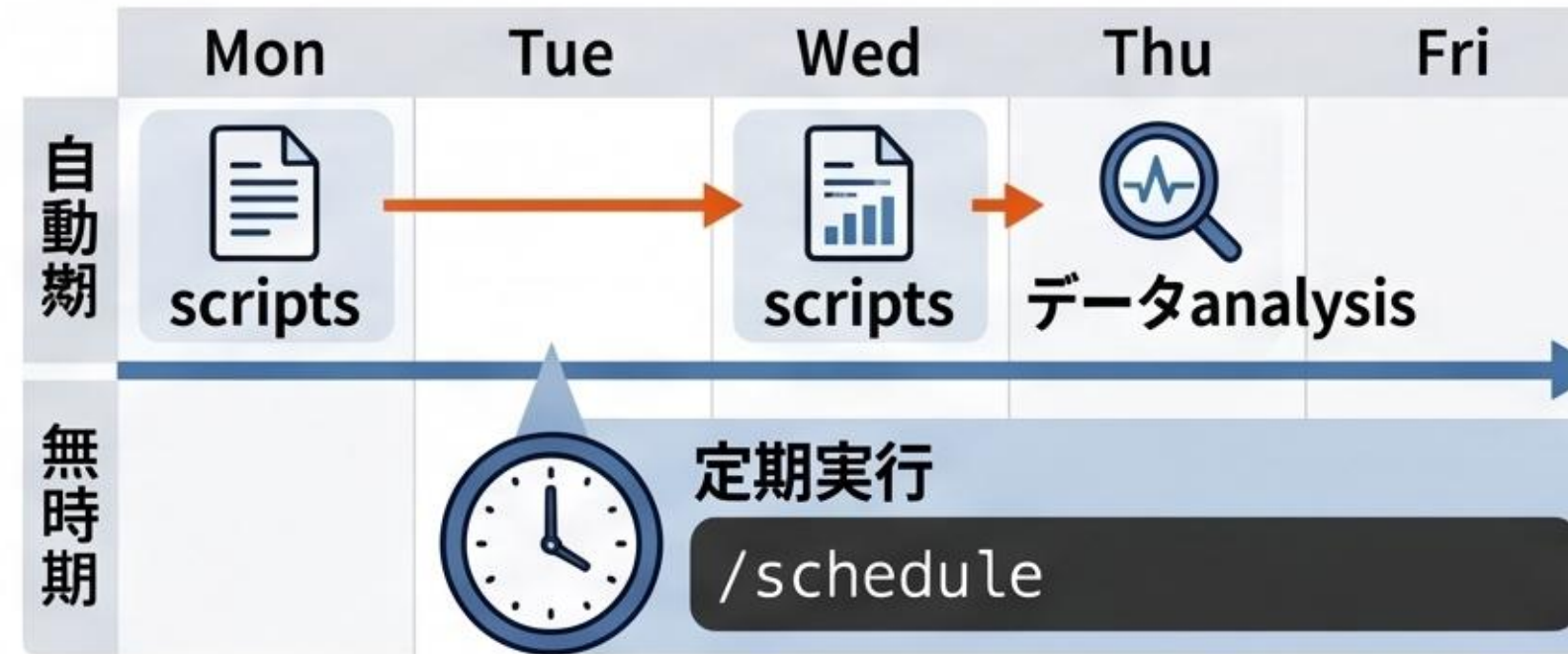


# 本日のトピック一覧

1 📄 Codex CLI 0.130.0 —  
`codex remote-control` でスマホ/  
外出先から自宅マシンを操作可能に



2 📄 Claude Code 「Routines」 —  
/schedule で定期実行 AI ジョブを 1  
行設定



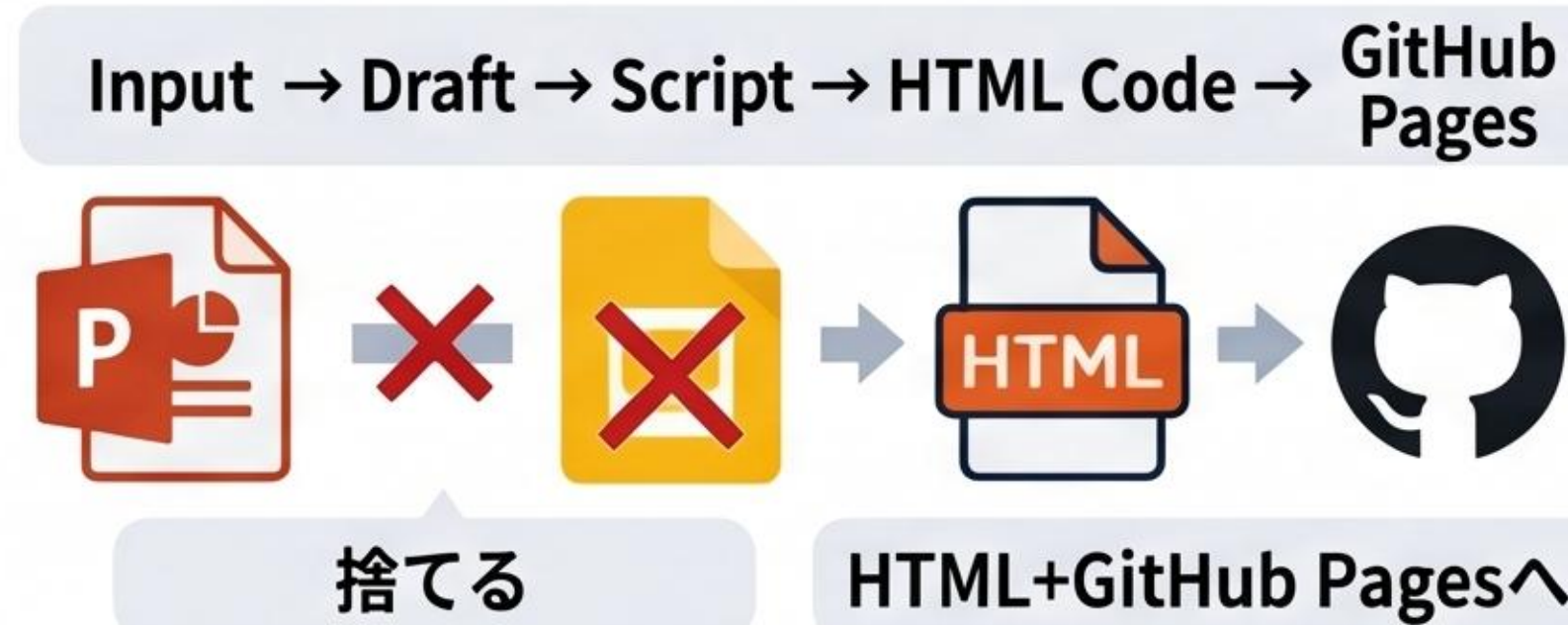
3 📄 16人エンジニアチームが  
Anthropic 解約 → Codex+Cursor へ  
— トークン効率議論が再燃



3 📄 16人エンジニアチームが  
Anthropic解約 → Codex+Cursor へ  
— トークン効率議論が再燃



4 📄 OneBiz: Claude Code で  
プレゼン資料を 5 ステップで作る  
— PowerPoint/Slides を捨てる



5 📄 Chormex に GPT-Realtime-2 で  
リアルタイム翻訳 + Codex で同時要約  
— 「ブラウザ = リアルタイム AI OS」

