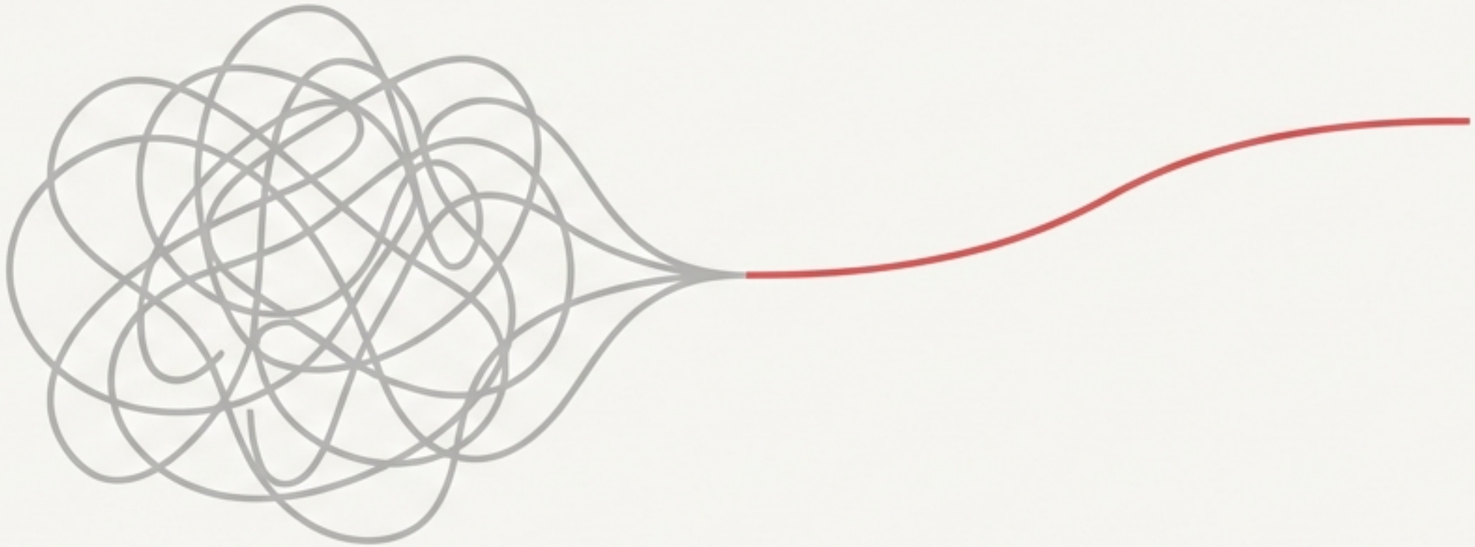
AI駆動開発がもたらす、開発組織の変革

Claude CodeとCodex CLIの組み合わせで実現する、スピードと品質の両立



bloom株式会社 AIX事業部 事例紹介

NotebookLM

「開発リソースが足りない」– こんな課題はありませんか？

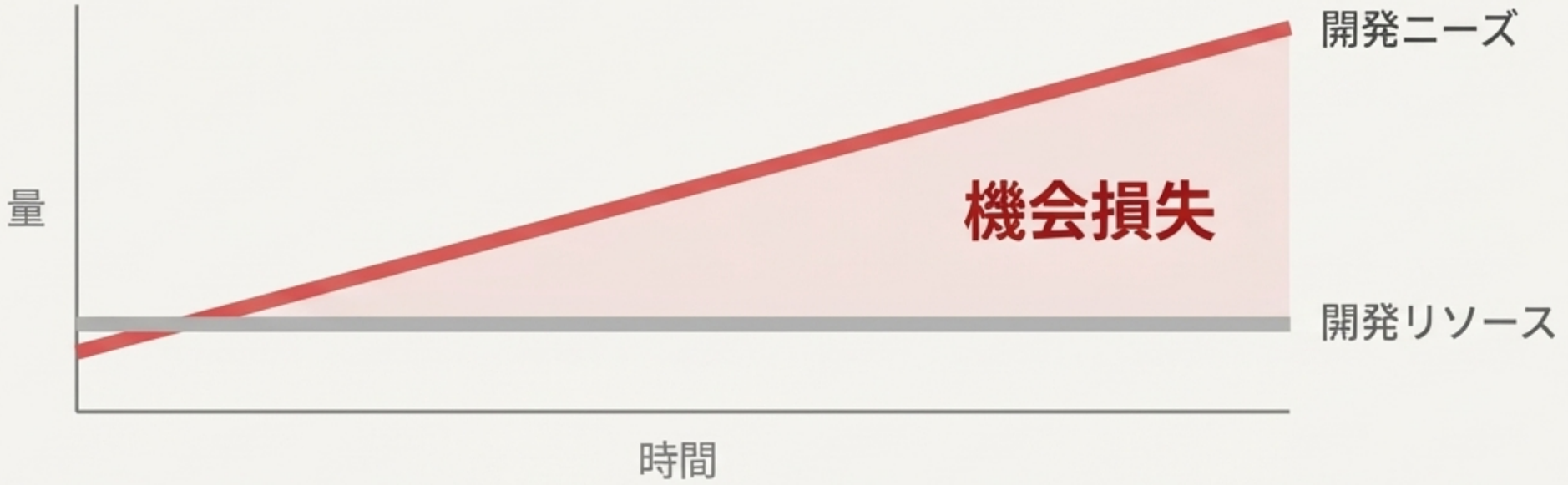
- 開発したいシステムはあるが、エンジニアが足りない。
- 社内にプログラミングの知見がなく、外注に頼らざるを得ない。
- エンジニアの採用が難しく、開発体制の強化が進まない。
- AIコーディングツールを使いたいが、セキュリティ面が心配。
- AIに任せきりにするのは不安で、品質をどう担保すべきかわからない。

これらの課題は、多くの企業が直面する共通の悩みです。本資料では、これらの課題を解決した具体的なアプローチをご紹介します。

NotebookLM

事業拡大の裏で深刻化していた、3つの開発課題

AIX事業部では複数の社内システム開発が同時進行する一方、事業の成長スピードに
開発リソースが追いつかないジレンマを抱えていました。



1. リソースの壁

開発ニーズは日々増加するも、限られたリソースでは着手できないプロジェクトが多数存在。

2. 品質の壁

エンジニアのスキルにばらつきがあり、コードレビューの負荷が増大。品質の維持が属人的に。

3. コストとノウハウの壁

外注は費用とコミュニケーションコストがかさみ、社内に知見が蓄積されない。

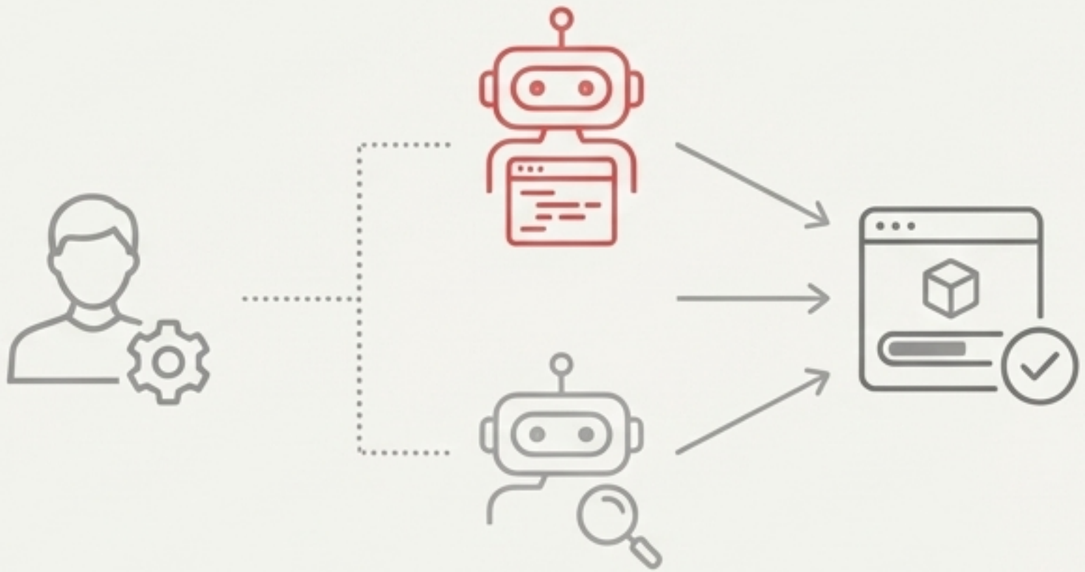
NotebookLM

課題解決の鍵は「AI駆動開発」という新たなアプローチ

AIを単なるツールではなく、「もう一人のエンジニア」として開発プロセスに組み込む。

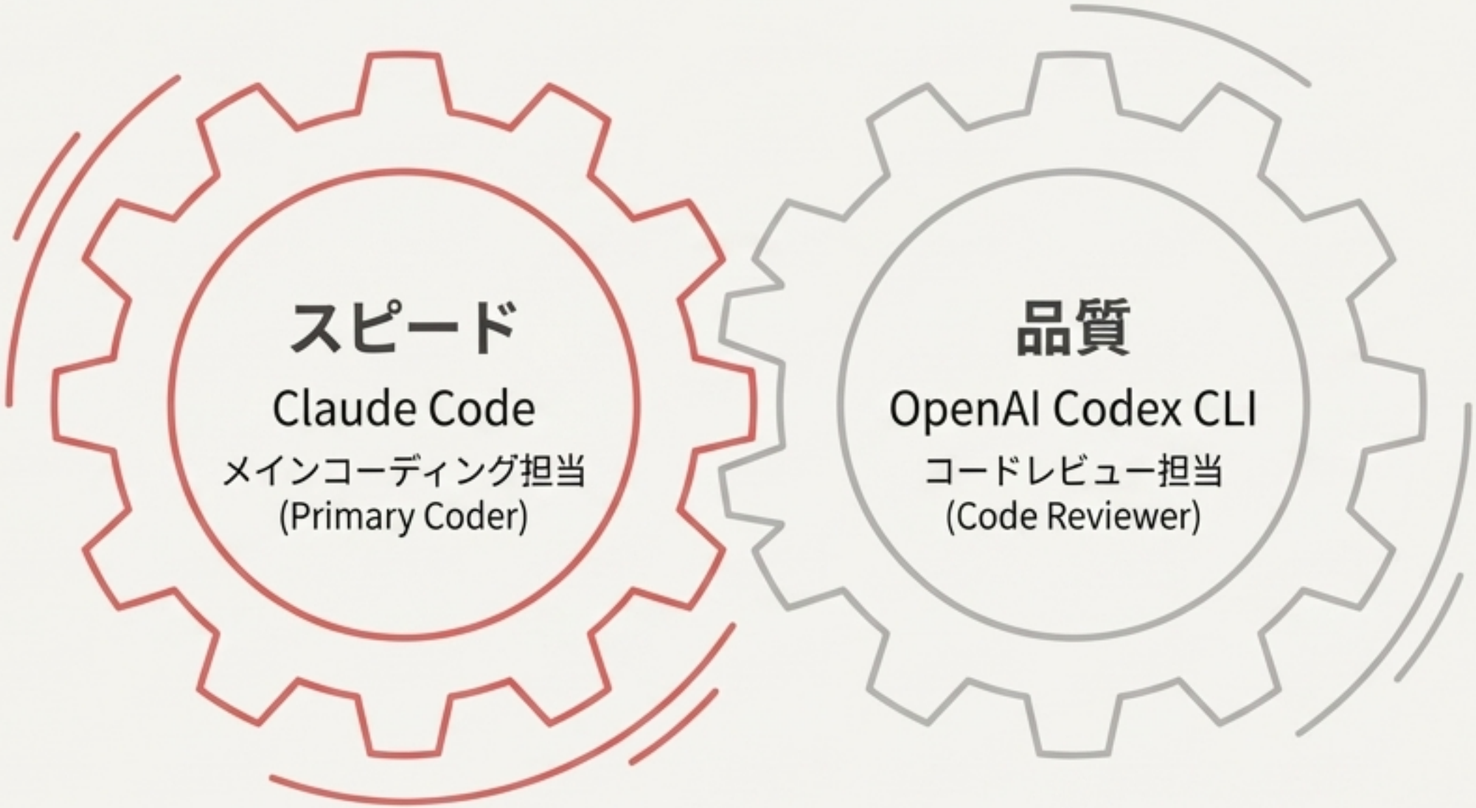
人間のエンジニアがAIに指示を出し、AIが高速にコードを生成し、別のAIがそれをレビューする。この協調体制により、人間の判断力とAIの処理能力を組み合わせ、開発のボトルネックを解消します。

従来、エンジニアの手が空くのを待つしかなかったプロジェクトに着手可能となり、組織全体の開発アジリティを向上させる。



NotebookLM

スピードと品質を両立する「デュアルエンジン」モデル



特性の異なるAIツールを適材適所で組み合わせる。Claude Codeの「高速な生成能力」とCodex CLIの「慎重かつ論理的な検証能力」を役割分担させることで、単一ツールでは実現できないスピードと品質の両立を可能にしました。

NotebookLM

なぜ「スピード担当」にClaude Codeを選んだのか

キー属性: ターミナル上で動作し、既存のワークフローに自然に組み込める高速コーディングエージェント。



圧倒的なコーディング速度

人間が書くよりも数倍速いペースでコードを生成。反復的な修正作業もストレスなく実行。



開発環境との高い親和性

ターミナルと直接連携し、ファイル操作やgit、gcpコマンドの実行までAIに委任可能。



エンタープライズ級のセキュリティ

TeamプランはSOC 2認証を取得。入力データがモデル学習に使用されず、業務で安心して利用できる。

NotebookLM

なぜ「品質担当」にCodex CLIを任せたのか

キー属性: 高度なコーディング能力と論理的な整合性検証に優れ、レビュー用途に最適なエージェント。



複雑なロジックの検証能力

変数のスコープや型の不整合など、人間が見落としがちな細かい問題を的確に検出。



エッジケースの指摘

境界値の処理やエラーハンドリングの漏れなど、コードの堅牢性を高めるための重要なフィードバックを提供。



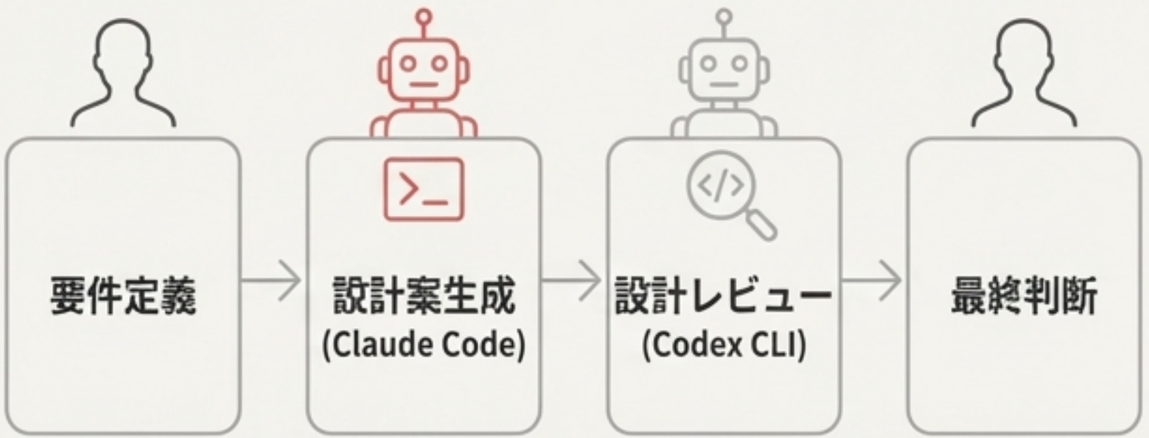
信頼できるセキュリティ

Claude Code同様、TeamプランでSOC 2認証を取得しており、セキュアな業務利用の基準をクリア。

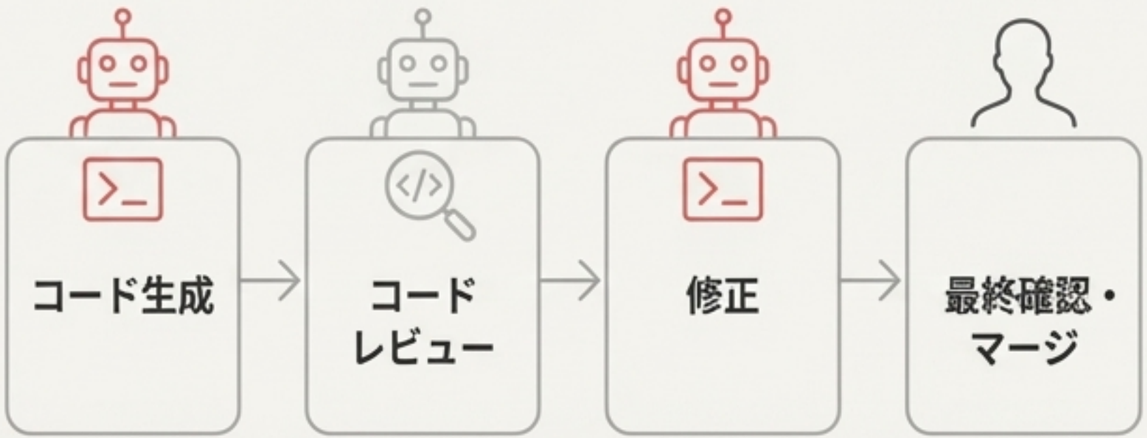
NotebookLM

設計から実装まで。AIを組み込んだ新しい開発ライフサイクル

設計フェーズ (Design Phase)



開発フェーズ (Development Phase)



NotebookLM

AI活用で最も重要な「セキュリティ」をどう担保したか

業務利用において、セキュリティと情報管理は最優先事項です。



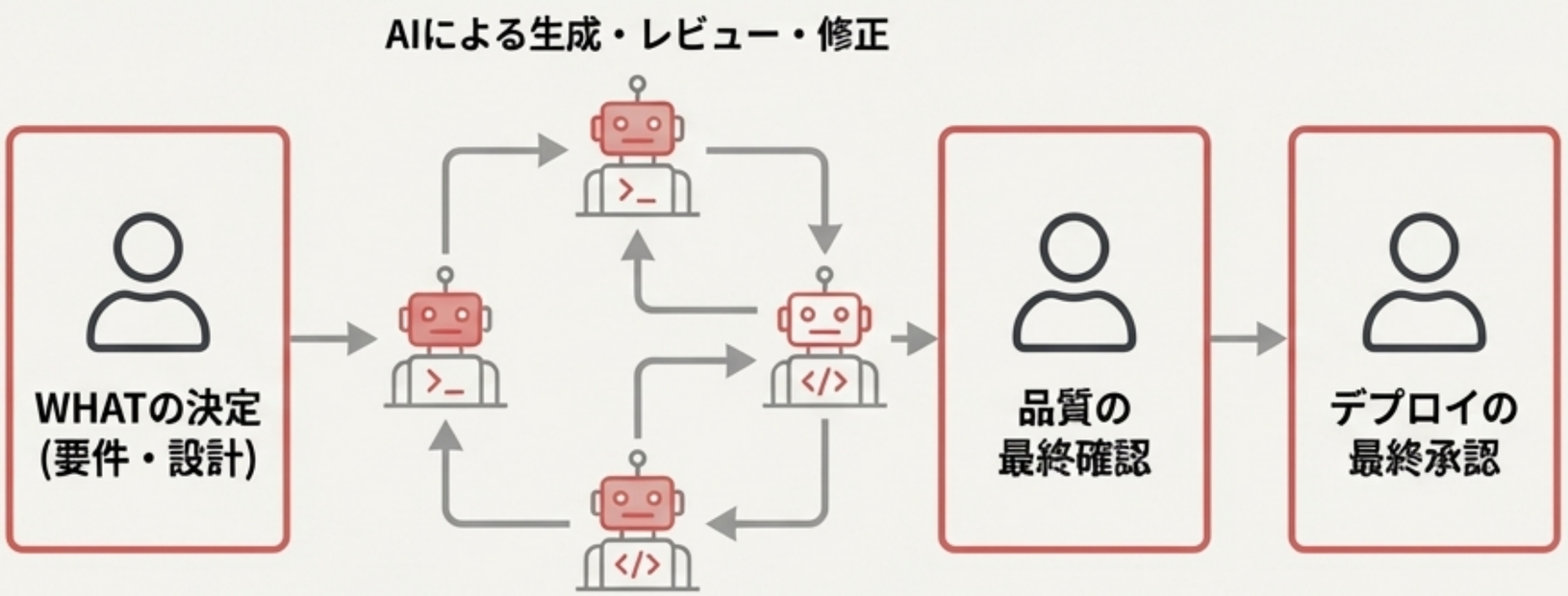
Our Security Measures

- **Teamプランの徹底活用**: Claude Code, Codex CLIともにTeamプランを契約。入力データがAIモデルの学習に利用されないことを保証。
- **第三者機関による認証**: 両サービスともSOC 2認証を取得済み。信頼性の高いセキュリティ管理体制が検証されている。
- **厳格な運用ルールの策定**: 社内ポリシーに基づき、入力してよいデータ範囲や確認プロセスを明文化した利用ガイドラインを策定・遵守。
- **アクセス権限の管理**: GCPの組織ポリシーを厳格に設定し、AIが実行できる操作範囲を制限。意図しない操作のリスクを低減。

NotebookLM

AIはあくまでパートナー。最終的な意思決定は人間が行う

AI駆動開発において最も重要な原則は、「本番環境に反映する責任は人間にある」という責任の所在を明確にすることです。



人間が主導権を握る3つのポイント

1. **WHATの決定**
   何を作るか（要件・設計）は、常に人間が主導権を持つ。

2. **品質の最終確認**
   Codex CLIのレビューに加え、人間の目によるコード確認と動作検証を必須とする。

3. **デプロイの最終承認**
   本番環境への反映は、必ず人間が最終的な承認を行う。

NotebookLM

成果: 開発工数を50〜70%削減

50-70%

実装フェーズの時間が大幅に短縮され、プロジェクトの回転率が向上。

従来は一行ずつコードを書いていた部分がAIによる一括生成に置き換わり、非エンジニアでもプロトタイプ開発が可能になるなど、組織全体の生産性が飛躍的に向上しました。

NotebookLM

数字以上の価値。組織にもたらされた3つの質的変化



1. 開発の民主化

非エンジニアでもAIの支援を受け、一定のアプリケーション開発が可能に。アイデアをすぐに形にできる文化が醸成。



2. 付加価値への集中

削減できた時間を要件の精査やユーザーテストに再配分。開発全体の質が向上。



3. 開発者のスキル向上

AIが生成するベストプラクティスに日常的に触れることで、開発者自身の技術知識が格段に向上。

NotebookLM

導入の過程で直面した壁と、それを乗り越えた工夫

Challenge 1: AIの強力な権限管理



課題

Claude Codeはファイル操作やコマンド実行など権限が強く、意図しない操作のリスクがあった。

対策

実行前の確認を徹底する運用ルールを策定。GCPの組織ポリシーで実行可能な操作範囲を厳格に制限。

Challenge 2: 複数ツールの最適な使い分け



課題

当初は役割分担が曖昧で、かえって効率が上がらない時期があった。

対策

試行錯誤を重ね、各ツールの特性を深く理解。「スピードのClaude、品質のCodex」という明確な役割分担を確立。

Key Takeaway

異なるモデルの得意領域を掛け合わせる相互チェック体制が、人間のレビュー負荷を軽減し、コードの堅牢性を高める上で極めて重要だった。

NotebookLM

このアプローチは、どのようなチームに有効か

こんな企業・チームにおすすめ

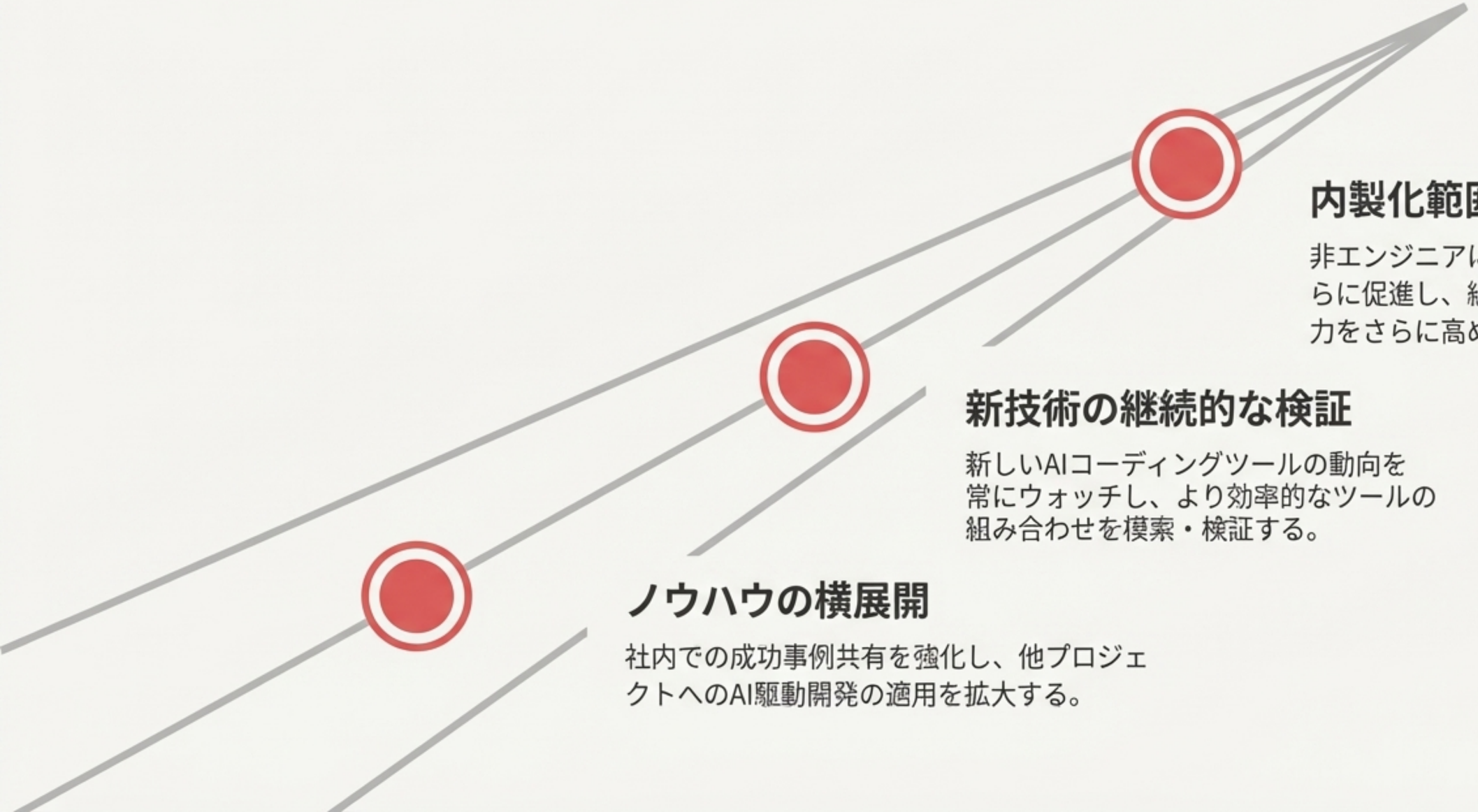
- ✓ 開発リソースが不足し、プロジェクトが滞っている。
- ✓ エンジニア以外のメンバーも開発に参加できる体制を作りたい。
- ✓ セキュリティと品質を担保しながら、AIコーディングツールを導入したい。

成功の要点

1. **ツールの役割分担:** Claude Codeの「速度」とCodex CLIの「品質」を両立させたこと。
2. **セキュリティの担保:** TeamプランとSOC 2認証サービスを利用し、信頼性を確保したこと。
3. **人間の最終判断:** すべての工程で人間が最終的な意思決定を行う体制を徹底したこと。

NotebookLM

AI駆動開発の進化に向けた、今後の展望



内製化範囲の拡大

非エンジニアによる開発参加をさらに促進し、組織全体の内製化能力をさらに高めていく。

新技術の継続的な検証

新しいAIコーディングツールの動向を常にウォッチし、より効率的なツールの組み合わせを模索・検証する。

ノウハウの横展開

社内での成功事例共有を強化し、他プロジェクトへのAI駆動開発の適用を拡大する。

NotebookLM