資料１　クロード・ソネット4.5に関するブリーフィング・ドキュメント

エグゼクティブ・サマリー

Anthropicは、最新のフロンティアモデルである **Claude Sonnet 4.5** を発表しました。このモデルは、**コーディング、複雑なエージェント構築、コンピュータ利用** において世界最高水準の性能を達成し、**推論と数学能力** においても大幅な向上を示しています。
Sonnet 4.5は、**SWE-bench Verified** や **OSWorld** などの主要ベンチマークで最先端のスコアを記録し、**30時間以上の複雑タスクでも集中力を維持** する能力が確認されています。

このリリースには、モデル自体の性能向上に加え、**製品エコシステム全体にわたる大規模なアップグレード** が含まれます。
開発者向けには、AnthropicがClaude Codeで使用しているインフラ **Claude Agent SDK** が公開され、独自のAIエージェント構築が可能になります。
また、**チェックポイント機能やVS Code拡張機能** がClaude Codeに追加され、**Claude APIにはコンテキスト編集機能とメモリツール** が導入されました。

安全性とアライメント面でも、Sonnet 4.5はこれまでで最もアライメントが取れたフロンティアモデルです。
媚びへつらい、欺瞞、権力希求 といった問題行動が大幅に減少し、**プロンプトインジェクション攻撃への耐性** も向上。
本モデルは **AIセーフティレベル3（ASL-3）** の保護下でリリースされ、危険コンテンツを検知する分類器が実装されています。

Claude Sonnet 4.5は**即日API経由で利用可能**で、価格はClaude Sonnet 4から据え置き。
性能が大幅に向上した**ドロップイン・リプレースメント**として、すべてのユーザーにアップグレードが推奨されています。

中心的な能力とパフォーマンス

フロンティア・インテリジェンス

Claude Sonnet 4.5は、現代の業務に不可欠な3つの主要領域で最高水準の能力を発揮します。

- **コーディング:** 世界最高のコーディングモデル
- **エージェント構築:** 複雑なエージェントを構築するための最強モデル
- **コンピュータ利用:** コンピュータを最も効果的に活用できるモデル

実用面では、**30時間以上の長時間タスク**でも集中を維持できることが確認されています。
専門家による評価でも、旧モデル（Opus 4.1含む）に比べ、**専門分野に特化した知識と推論能力が劇的に向上**しています。

ベンチマークにおける成果

ベンチマーク	説明	Sonnet 4.5のスコア	備考
SWE-bench Verified	実世界のソフトウェアコーディング能力	77.2%	10回平均。高計算構成では82.0%達成
OSWorld	実世界のコンピュータタスク実行能力	61.4%	Sonnet 4の42.2%から大幅向上
AIME	数学・推論能力	向上	温度1.0でサンプリング
MMMLU（非英語）	14言語での多タスク言語理解	向上	5回実行の平均
Finance Agent	金融分析タスク	向上	Vals AIリーダーボードより

顧客およびパートナーによる評価

- **GitHub Copilot:**「多段階推論とコード理解で著しい改善が見られた。」
- **Devin:**「計画性能18%、総合評価12%向上。Sonnet 3.6以来最大の飛躍。」
- **Canva:**「最も複雑なタスクで目覚ましい成果を上げた。」
- **Cursor:**「長期タスクでの改善が著しい。」
- **Perplexity:**「脆弱性対応時間を44%短縮、精度25%向上。」
- **顧客事例（コーディング）:**「30時間以上の自律コーディングが可能。」
- **顧客事例（金融分析）:**「人間レビューを減らせる投資グレードの洞察。」
- **顧客事例（コード編集）:**「エラー率が9%→0%に。」

製品エコシステムのアップグレード

Claude Sonnet 4.5 モデル

- **提供:** APIにて `claude-sonnet-4-5` を指定して利用可能
- **価格:** Claude Sonnet 4と同一（入力 $3/Mトークン、出力 $15/Mトークン）

開発者向けツール

- **Claude Agent SDK:**
  メモリ管理、許可システム、サブエージェント連携を解決するSDK。
  コーディング以外の多様なタスク対応も可能。
- **API強化:**
  - コンテキスト編集機能
  - メモリツール

製品機能強化

- **Claude Code:**
  - チェックポイント機能
  - ターミナル刷新
  - ネイティブVS Code拡張

- **Claude Apps:**
  - コード実行・ファイル作成機能（有料プラン）
- **Claude for Chrome:**
  Maxユーザー向けに提供開始

研究プレビュー：「Imagine with Claude」

- **概要:** Claude Sonnet 4.5の能力を実証する研究プレビュー。
  対話に応じてリアルタイムにソフトウェアを生成する実験機能。
- **提供:** Maxサブスクライバー向けに5日間限定公開（claude.ai/imagine）

安全性とアライメント

最もアライメントが取れたフロンティアモデル

- **問題行動の低減:** 媚びへつらい、欺瞞、権力希求などの減少
- **安全性評価:** メカニスティック・インタプリタビリティを用いたテストを導入
- **プロンプトインジェクション対策:** 防御を大幅に改善

AIセーフティレベル3（ASL-3）保護

- **分類器:** CBRN関連の危険な入出力を検知・遮断
- **誤検知:** Claude Opus 4比で1/2に減少
- **許可リスト:** 生物学研究・サイバーセキュリティ分野の顧客に提供

結論と推奨事項

Claude Sonnet 4.5は、**コーディング・エージェント・コンピュータ利用で業界最高性能**を達成したモデルです。
Claude Agent SDK の公開により、AIアプリケーション開発の新たな可能性が開かれました。

Anthropicは、**旧価格で大幅な性能向上**を実現したSonnet 4.5へのアップグレードをすべてのユーザーに推奨しています。
詳細は **公式システムカード・モデルページ・ドキュメント** を参照してください。

資料２　AIエージェントの構築とコンテキストエンジニアリングに関するブリーフィング

エグゼクティブサマリー

本ブリーフィングは、**効果的なAIエージェントシステム構築**に関する最新知見を統合したものです。
成功の鍵は **複雑なフレームワークを避け、シンプルで構成可能なパターンを採用すること** にあります。

主な結論

1. **シンプルさが成功の鍵:**
   不要な抽象化を避け、理解しやすい構成要素を採用。
2. **コンテキストは有限な資源:**
   「注意力の予算」として慎重に管理。
3. **ジャストインタイム情報取得:**
   必要時に動的に情報を取得する方式。
4. **長期タスク戦略:**
   要約・外部メモリ・サブエージェントを活用。
5. **ツール設計の重要性:**
   明確・重複なし・誤用されにくいツールが鍵。

最も効果的なエージェントは「最も洗練された」ものではなく、**特定のニーズに適したシステム**である。

---

1. 効果的なAIエージェントの構築

1.1 エージェントシステムの分類

タイプ	説明	最適な用途
ワークフロー	LLMとツールが事前定義パスで連携	予測可能性と一貫性が必要なタスク
エージェント	LLMが自律的にツールを使用し制御	柔軟性と動的判断が求められるタスク

多くのアプリは単一のLLM呼び出しで十分。
エージェント導入はコスト増に見合う場合のみ有効です。

1.2 構築パターンとワークフロー

- **拡張LLM:** 検索・ツール・メモリで強化された基本単位
- **プロンプト連鎖:** サブタスクに分割し順次処理
- **ルーティング:** 入力を分類しタスク振り分け
- **並列化:** 複数LLMが同時に作業
  - セクショニング: サブタスクを並行実行
  - 投票: 複数出力から最良を選定
- **オーケストレーター・ワーカー:** 中央LLMがサブタスクを分配
- **評価・最適化:** 応答→評価→改善ループ
- **自律型エージェント:** 環境からのフィードバックで自律動作

1.3 実用的な応用例

- **カスタマーサポート:** 会話＋ツール操作統合
- **コーディングエージェント:** 自動テストと評価ループが可能

---

2. コンテキストエンジニアリング：次世代のプロンプト技術

2.1 コンテキストが重要である理由

- **コンテキストの陳腐化:** トークン増加で想起精度が低下
- **注意力の予算:** トークンごとに注意リソースを消費
- **アーキテクチャ制約:** Transformerは n^2 関係を処理するため長文で性能低下

→ **「最小限で高シグナルなトークンセット」** を設計することが重要。

2.2 効果的なコンテキストの構成要素

- **システムプロンプト:**
  明確で直接的な指示を使用し、柔軟性を保つ。
  （※文書続きあり）
