

Codex-Spark 詳細分析レポート

エグゼクティブサマリー

本調査では、OpenAI ¹ が公開した「GPT-5.3-Codex-Spark（以下、Codex-Spark）」について、一次情報（公式ブログ、公式ドキュメント、公式ポリシー）を中心に、仕様・技術・性能・運用上の評判・リスクを整理し、技術者および意思決定者向けに導入判断材料を提示する。²

Codex-Sparkは「リアルタイム・コーディング専用」を明確に打ち出した研究プレビューであり、**秒間1,000トークン超の生成速度と、WebSocket・Responses API最適化によるレイテンシ削減（ラウンドトリップ80%削減、TTFT 50%改善等）**という「体感速度」を最優先にした設計が最大の特徴である。³
一方で、公開ベンチマークは「強い性能」までで**数値は未指定**であり、また一般的なAPI提供は「ローンチ時点では不可（ただし一部デザインパートナーに限定提供）」という制約がある。⁴

セキュリティ面では、Codex全体としてサンドボックス・承認（Approval）・ネットワーク制御・監査（OTel）などの統制が用意され、人的統制と環境隔離を前提とした「エージェント運用」設計が進んでいる。⁵

ただし、運用コミュニティでは「実際にどのモデルがサーブされたか」の透明性（サイレントなルーティング／表記ずれ）に関する懸念が既に観測され、監査・検証設計が重要になる。⁶

主要発見の箇条書き（要点優先）

Codex-Sparkは、研究プレビューとして **テキスト専用・コンテキスト128k** で提供開始（高需要時はキューイング可能性）と明示。⁷

推論・配信経路の最適化が重視され、**WebSocket常時接続＋推論スタック改修**でオーバーヘッド・TTFTを大幅削減（数値が公開されている点は評価可能）。³

ハードウェアはCerebras ⁸ のWSE-3を用いた「低遅延サービング層」を構築。今後の増強ロードマップ（段階的統合、容量増強）も公式に示唆。⁹

ベンチマークは「SWE-Bench Pro／Terminal-Benchで強い」「時間はGPT-5.3-Codexの一部」と“定性的”に述べられるが、スコア自体は**未指定**。³

ローンチ時点で一般APIには非提供（ただし「一部デザインパートナー」に限定してAPI提供を開始）。定番のAPI自動化用途は、当面 gpt-5.2-codex を推奨する、と公式に明確化。⁴

安全性について、Codex-Sparkはメインラインと同等の安全訓練（サイバー関連含む）を施し、Prep Framework上の高能力閾値に到達する蓋然性は「あり得ない（plausible chanceなし）」と評価したと公式表明。³

運用現場では速度への期待が大きい一方、「モデル表記と実サーブの不一致」や「サイレントなルーティング懸念」が具体例として議論され、監査ログ・検証手段の整備が必須。¹⁰

調査設計

調査対象

対象モデル：GPT-5.3-Codex-Spark（Codex-Spark）。⁷

比較・参照：同系列のGPT-5.3-Codex（長時間タスク向け）、gpt-5.2-codex（当面のAPI推奨）、主要競合のコーディング／エージェント向けモデル（Claude Opus系、Gemini系）。¹¹

評価軸（明示）

公式仕様：リリースノート、提供形態（アプリ／CLI／IDE／API）、料金、ライセンス／規約、モデルサイズ、トレーニングデータ概要。¹²

技術的特徴：推論速度・レイテンシ、メモリ要件（ユーザー側／提供側）、最適化（WebSocket、推論スタック、セッション初期化）、量子化／蒸留の有無。¹³

性能評価：言語理解・生成品質・コーディング能力（既知ベンチ）、安全性（有害出力、デュアルユース評価）、マルチモーダル対応。¹⁴

比較：主要競合との定量・定性比較（表）。ベンチ差は測定環境で変動し得る点を前提に扱う。¹⁵

実運用評判：開発者フォーラム／SNS／企業導入（一次情報優先）、障害・品質懸念、コスト対効果。¹⁶
セキュリティ・プライバシー・法務：サンドボックス、ネットワーク、監査、データ学習、利用規約／ポリシー、ライセンス混入リスク。¹⁷

公式仕様

公式仕様サマリー（未指定は明記）

項目	公式情報（要約）	根拠
リリース形態	研究プレビュー（research preview）	7
リリース日	二月中旬（公式ブログ／Codex changelogに日付明示）	7
提供チャンネル	Pro向けにCodexアプリ／CLI／VS Code拡張で展開	4
モデル識別子	<code>gpt-5.3-codex-spark</code> （CLIで指定可能）	18
コンテキスト長	128k	7
入出力モダリティ	テキストのみ（text-only）	7
料金（サブスク）	Proは月額200ドルでアクセス提供（Codex pricingに明記）	19
料金（API従量）	未指定 （一般APIは「ローンチ時点で利用不可」。一部デザインパートナーに限定提供）	4
ライセンス／利用条件	クローズドなサービス提供（利用規約・Usage Policiesに従う）。Business/APIは別契約枠組み	20
モデルサイズ（パラメータ等）	未指定 （公式に開示なし）	3
トレーニングデータ概要	個別データ詳細は 未指定 。OpenAIは一般論として「公開情報、第三者提携データ、研究者提供/生成情報」等を挙げ、Business/APIは「データを既定で学習に使わない」と明記	21

リリースノート（運用者向けに要点）

Codex changelogには、Codex-Sparkの位置づけ（GPT-5.3-Codexの小型版・リアルタイム用途・128k・text-only）と、モデル切替手順（CLI/IDE/アプリのセレクト）が明示されている。²²

CLIでは以下のようにモデルスラグ指定で新規スレッドを開始できる（実運用ではログ監査の観点でも有用）。²²

```
codex --model gpt-5.3-codex-spark
# または
codex -m gpt-5.3-codex-spark
```

API提供の現状整理（矛盾ではなく「スコープ差」と解釈）

公式ブログでは「少数のデザインパートナーにAPI提供を開始」と述べる一方、Codex changelogおよびCodex pricingは「ローンチ時点で一般API提供なし（API-key運用はgpt-5.2-codex継続）」と明記している。

⁴ 従って現時点の実務解釈は「一般利用者にとってはAPI未提供、限定先行枠のみAPI提供」となる（デザインパートナー企業名は未指定）。⁷

技術的特徴とアーキテクチャ

レイテンシ最優先の設計思想

Codex-Sparkは、長時間の自律実行よりも「割り込み・リダイレクトしながら短いサイクルで反復」するリアルタイム開発に最適化され、「最小で狙い撃ちの編集」「頼られない限りテストを回さない」という軽量のデフォルト挙動を公式に明示している。³

この思想は、エディタ内の修正ループ（編集→確認→再編集）を高速化し、Vibe Coding的な探索（微修正の連続）に向く一方、品質保証（自動テスト・静的解析）を自動で“厚め”に回したい組織ではプロンプト設計やガードレールが必須になる。²³

推論速度とパイプライン最適化（定量開示がある部分）

モデル自体の生成速度として「1,000 tokens/s超」を掲げるだけでなく、エンドツーエンドでの体感速度を改善するため、配信経路・推論スタック・セッション初期化を作り直し、以下の改善を行ったと公式に数値開示している。³

- ・クライアント／サーバ往復オーバーヘッド：80%削減
- ・per-tokenオーバーヘッド：30%削減
- ・time-to-first-token（TTFT）：50%改善
- ・永続WebSocket接続：Codex-Sparkでデフォルト有効（将来的に全モデルのデフォルトへ）³

この種の「推論そのものの遅延」を数値で追い込み、プロトコル（WebSocket）を含めて最適化するアプローチは、長文生成の平均スループットだけでは到達しにくい“対話の瞬発力”を改善する、実務的に重要な方向性である（特にIDE内の体験）。³

低遅延ハードウェア層

Codex-SparkはCerebras²⁴のWSE-3上で稼働し、Codexの推論スタック内に「レイテンシ最優先のサービング層」を追加したと説明されている。²⁵

加えて、OpenAIは同社との提携により「超低遅延AI計算資源（750MW）を段階的に統合し、複数年にわたり増強」と述べており、単発のPoCでなく中長期の供給計画を伴う点が示唆される。²⁶

WSE-3の物理仕様（例：トランジスタ規模）などの詳細はOpenAI一次情報ではなく報道に依存する部分があ

り、例えば報道では「4兆トランジスタ」とされるが、ここは一次情報ではないため留保が必要である（参考：TechCrunch報道）。 27

メモリ要件・量子化／蒸留

- ・メモリ要件（ユーザー側）：クラウド提供が前提のため、GPUメモリ等の自己ホスト要件は**未指定**（ユーザーが主に管理すべきは、コンテキスト設計・履歴・ログ・コード資産の扱い）。 23
- ・量子化／蒸留：Codex-Sparkは「小型版」と説明されるが、圧縮手法（蒸留・量子化・アーキテクチャ縮小のいずれか/組合せ）は**未指定**であり、現時点で推測はできない。 3

性能評価とベンチマーク

コーディング能力（Codex-Spark）

公式は、SWE-Bench ProおよびTerminal-Benchにおいて「強い性能を示しつつ、GPT-5.3-Codexより短時間で完了」と述べ、所要時間の推定式（生成・prefill・ツール・ネットワークの合算）まで示すが、**スコアそのものは未指定**である。 3

したがって、現時点で定量評価に使えるのは「速度（tokens/s）」「パイプライン改善量」「text-only/128k」といった運用パラメータであり、品質スコアは（外部独立評価が出るまで）相対比較が難しい。 28

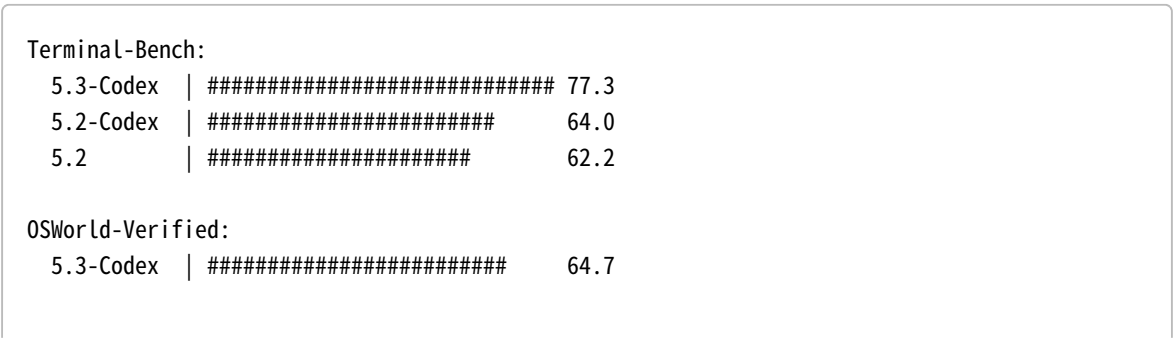
参照としてのGPT-5.3-Codex（公式ベンチ）

Codex-Sparkの品質を間接的に把握するには、親モデル（GPT-5.3-Codex）の公開ベンチを参照するのが現実的である。OpenAI公式ブログは、以下のベンチ数値（xhigh）を提示している。 29

指標（公式ブログ）	GPT-5.3-Codex	GPT-5.2-Codex	GPT-5.2
SWE-Bench Pro（Public）	56.8%	56.4%	55.6%
Terminal-Bench	77.3%	64.0%	62.2%
OSWorld-Verified	64.7%	38.2%	37.9%
GDPval（wins or ties）	70.9%	未指定	70.9%
Cyber CTF	77.6%	67.4%	67.7%
SWE-Lancer IC Diamond	81.4%	76.0%	74.6%

29

簡易チャート（棒の長さは相対・目安）：



5.2-Codex	#####	38.2
5.2	#####	37.9

(数値はOpenAI公式ブログ掲載値) 29

なお、GPT-5.3-CodexのOSWorld-Verified説明では「モデルがvisionを用いてタスクを完了」と明記されており、少なくとも評価設定上は視覚入力を扱う前提がある。一方、Codex-Sparkは現時点でtext-onlyと明記されているため、同種の“画面理解＋操作”系の適性は将来拡張待ちとなる。 30

安全性・有害出力（Codex-SparkとGPT-5.3-Codexの位置づけ差）

Codex-SparkについてOpenAIは「メインラインと同等の安全訓練（サイバー関連含む）を行い、標準デプロイ評価の結果、Prep Framework上の高能力閾値（サイバー/生物）に到達する蓋然性はない」と述べる。 3

対照的にGPT-5.3-Codexは、システムカードで「サイバー領域でHigh capabilityとして扱う最初のローンチ」と記述され、閾値到達の確証はないが予防的に高能力扱いでセーフガードを有効化すると説明している。 31

有害出力のベースライン評価として、GPT-5.3-Codexシステムカードは会話的な“disallowed content”評価の一部（Production Benchmarks）を表として公開している（例：illicit violent activities等）。 32

ただし同カードは「この会話評価はコーディング特化モデルの実世界リスクを反映しない可能性」を述べており、スコアの解釈は“安全側の指標”として限定的に扱うべきである。 33

主要競合との比較

比較における注意点（測定環境差）

エージェント型コーディング評価は、同一モデルでも実行基盤やリソース制約でスコアが数ポイント変動し得ることが指摘されており、異社モデルのスコアを“単純比較”するのは危険である。 34

そのため本表では、**公式に確定している運用パラメータ（提供形態、コンテキスト、料金、モダリティ、API可否）を重視し、ベンチ値は「出典付きで併記」**するが、同一土俵での比較でないことを明示する。 35

主要モデル比較表

観点	Codex-Spark	GPT-5.3-Codex	gpt-5.2-codex	Claude Opus 4.6	Gemini 3 Pro Preview
提供元	OpenAI 1	OpenAI	OpenAI	Anthropic 36	Google 37
主用途	リアルタイム反復（低遅延） 3	長時間の推論・実行＋コンピュータ作業 29	API自動化の当面推奨（Codex側が言及） 38	長文知識作業・コーディング（開発者向け提供） 39	マルチモーダル理解・エージェント用途（API提供） 40
提供形態	Pro向けアプリ/CLI/IDE、APIは限定（一般は不可） 4	有料プランでCodex各面、APIは準備中 41	APIで利用可能（gpt-5.2系のAPI価格表あり） 42	Claude API/主要クラウド等（1M contextは自社プラットフォームβ等） 43	Gemini API/Vertex AI等（モデルIDと価格表が公開） 44

観点	Codex-Spark	GPT-5.3-Codex	gpt-5.2-codex	Claude Opus 4.6	Gemini 3 Pro Preview
コンテキスト	128k (text-only) ⁷	未指定（公式ブログ内に明示なし） ²⁹	未指定（本調査範囲の一次情報では特定不能）	最大1M（β、提供条件あり） ⁴⁵	1M / 64k (in/out) ⁴⁶
マルチモーダル	テキストのみ（将来マルチモーダル予定） ³	OSWorld説明ではvision利用の前提あり ²⁹	未指定	公式にマルチモーダル提供（価格表も画像入力言及） ⁴⁷	マルチモーダル（テキスト/画像等） ⁴⁸
速度	1000 tokens/s 超+TTFT/RTT改善 ³	25%高速化（Codex向け） ²⁹	未指定	未指定（本調査の一次情報ではtokens/sは非公開）	未指定（一次情報ではtokens/s非公開）
ベンチ（公表）	「強い性能」だがスコア未指定 ³	複数ベンチの%が公表 ²⁹	未指定	スコアの詳細は各社資料依存（公式はsystem card等） ⁴⁹	仕様・価格は明確、ベンチは一次情報で統一的に確認困難 ⁴⁶
価格	Pro月額200ドルに含まれる ¹⁹	Plus等に含まれる（Codex pricing） ¹⁹	API従量（例：GPT-5.2入力\$1.75/出力\$14/MTok） ⁵⁰	入力\$5/出力\$25 per MTok（長文はプレミアム条件あり） ⁵¹	入力\$2/出力\$12 per MTok（条件で変動） ⁴⁴
ライセンス/法務	OpenAI規約・ポリシーに従う（サービス提供） ⁵²	同左	同左	各社規約	各社規約

結論として、Codex-Sparkは「速度と体験」に振り切っており、**API統合やマルチモーダルは現段階で限定**、一方で「IDE内の反復効率」を最短で引き上げたい場合に差別化が大きい。 ⁵³

実運用での評判・導入状況

ポジティブな反応（速度体験への期待）

公式発表の文脈では「リアルタイム・コラボレーション」や「新しい対話パターン」が強調され、Cerebras側も「fast inferenceが可能にする新しい体験」を述べている（以下は短い引用）。 ²⁵

- 代表的引用（英→日、短く翻訳）：
“...discover what fast inference makes possible.”
→「高速推論で何が可能になるかを（開発者と共に）発見したい」 ³

コミュニティ側では「用途でモデルを使い分ける」視点があり、海外掲示板では「低遅延モデルは別枠の役割を持つ」という整理が見られる（以下は短い引用）。 ⁵⁴

- 代表的引用（英→日、短く翻訳）：
“you’ll use this for some things...and keep using the other model for heavier tasks.”
→「軽い用途はこれ、重い用途は別モデル、という使い分けになる」 ⁵⁵

ネガティブ／懸念（透明性・モデル同定・品質確認）

運用上の早期リスクとして目立つのは「ユーザーが選択したモデル名と、実際に応答オブジェクトに記録されるモデル名の不一致」や、「サイレントなルーティングを疑う声」である。 56

- 代表的引用（英→日、短く翻訳）：
“response.model reports a different model: gpt-5.2-2025-12-11.”
→「応答オブジェクト上は別モデル（5.2系）が返っている」 57

- 代表的引用（英→日、短く翻訳）：
“how do i actually verify this?... silent downgrades”
→「どう検証すればよい？サイレントに格下げされていないか不安」 58

この種の疑義は、研究プレビューや需要逼迫時のキューイング・レート調整がある状況で特に増幅しやすい。Codex-Spark側でも高需要時の制限／一時キューを明示しているため、“**体験が揺れる期間**”がある前提で運用設計すべきである。 59

企業導入事例

Codex-Sparkについて、公式に「デザインパートナーにAPI提供」と述べるが、企業名・導入事例・成果指標は**未指定**であり、現時点で一次情報として確認できない。 3

リスク・制限・回避策と推奨

セキュリティ・プライバシー

Codexのセキュリティは、(a) サンドボックス（技術的に何ができるか）と、(b) 承認ポリシー（いつ許可を求めるか）の二層で設計されており、ローカル／クラウド双方で「既定はネットワークOFF＋ワークスペース制限」を基本にしている。 60

ネットワークやWeb検索の有効化は利便性を上げる一方、**プロンプト注入、資格情報漏えい、ライセンス制約コードの混入**などのリスクが明示されているため、許可リスト方式・HTTPメソッド制限・レビューを前提にする必要がある。 61

データ取り扱いについては、Business/APIプラン等では「既定で学習に使わない」「入力と出力の所有（法令が許す範囲）」といったコミットメントが明記されているため、組織導入ではこの契約枠の採用が実務上の基本線になる。 62

ただし、個人向け利用は利用規約に従い、用途制限（Usage Policies）も適用されるため、社内利用ポリシーと整合させる必要がある。 52

法的懸念（要点）

- 出力物の扱い：Business/API側では「入力・出力の所有」に触れているが、法域や内容により例外があり得るため、知財・契約部門と運用ルール化が必要。 63
- デュアルユース：GPT-5.3-Codexはサイバー領域で高能力扱いとなり、本人確認ベースのTrusted Accessを試行している。Codex-Sparkは高能力閾値に達しない評価だが、サイバー関連の安全訓練を含むとされ、将来の運用制約が変わる可能性は残る。 64

既知の制限と回避策（運用に直結するもの）

制限（現状）	影響	回避策（現実解）	根拠
text-only	UI/画面理解が必要な作業は不向き	画面操作・視覚入力が必要なならGPT-5.3-Codexなどへ切替／将来の対応待ち	65
コンテキスト128k	巨大リポジトリの“一括投入”が難しい	AGENTS.mdや分割投入、要約（compaction）運用、必要箇所だけ注入	23
自動テスト非実行がデフォルト	バグ混入の確率が上がる	「必ずテストを実行して」「変更は最小」等をルール化、CI連携	23
高需要時の制限/キュー	作業が止まる・遅延	重要作業は別モデルへのフェイルオーバー、作業時間帯調整	59
一般API未提供	CI/CD自動化へ直結しない	APIキー運用は当面 gpt-5.2-codex、Sparkは人間の反復用に割り切る	66
モデル同定/透明性懸念	監査不能・信頼低下	ログ/OTelで <code>response.model</code> 等を保存し、バージョン固定と検証手順を整備	67

推奨（導入可否と運用上の注意点）

導入推奨（条件付き）：

- IDE内での反復（微修正・UI調整・局所リファクタ）を日常的に行い、開発者体験（待ち時間）をボトルネックとして認識している組織では、月額Pro枠の費用に対して生産性リターンが出やすい。 53
- ただし品質保証は自動化されないため、CI/レビュー手順の整備が前提となる（特に「勝手にテストを回さない」設計が明言されている点は重要）。 23

導入見送り／慎重推奨：

- すぐにAPI経由で組込みたい（CI上の自動修正、SaaS統合など）場合、Codex-Sparkは一般API未提供のため見送り。代替はgpt-5.2-codex等で設計し、将来のSpark API一般化を待つ。 66
- 監査要件が強い組織では、モデル同定の懸念例が出ているため、OTel/ログ保全・バージョン固定・検証テスト（既知タスクで差分確認）を“導入要件”として契約前に定義すべき。 68

導入判断フロー（簡易図）

flowchart TD

```

A[目的は何か] --> B{リアルタイム反復が主?}
B -->|Yes| C{テキスト中心?}
C -->|Yes| D[Codex-Sparkを検証導入]
C -->|No| E[視覚/画面操作が重要→別モデル/別製品]
B -->|No| F{長時間の自律実行/調査が主?}
F -->|Yes| G[GPT-5.3-Codex等で長時間タスク]
F -->|No| H[既存Codexモデルで十分]
D --> I{監査/セキュリティ要件}
I -->|厳格| J[サンドボックス/承認/OTel/ログを必須化]
I -->|通常| K[デフォルト設定で開始→段階的に権限拡張]

```


将来展望（公式に根拠がある範囲）

OpenAIは、Codex-Sparkを「超高速モデル群の最初」と位置づけ、将来的により大きなモデル、より長いコンテキスト、マルチモーダル入力を導入すると述べている。³

また、低遅延計算資源の拡張を段階的に進め、複数年で容量を追加するとしており、研究プレビューから本格運用へ移る余地がある。⁶⁹

一方で、需要逼迫による制限やキューが明示されていることから、短期的には“最高速の体験が常に保証される”とは限らず、運用上はフォールバック設計が現実的である。⁵³

参考になった主要ソース一覧（URL付き、一次情報優先）

■ OpenAI（一次情報）

<https://openai.com/index/introducing-gpt-5-3-codex-spark/>
<https://developers.openai.com/codex/changelog/>
<https://developers.openai.com/codex/pricing/>
<https://developers.openai.com/codex/models/>
<https://openai.com/index/introducing-gpt-5-3-codex/>
<https://openai.com/index/gpt-5-3-codex-system-card/>
<https://cdn.openai.com/pdf/23eca107-a9b1-4d2c-b156-7deb4fbc697c/GPT-5-3-Codex-System-Card-02.pdf>
<https://openai.com/index/trusted-access-for-cyber/>
<https://openai.com/index/cerebras-partnership/>
<https://openai.com/enterprise-privacy/>
<https://openai.com/policies/usage-policies/>
<https://openai.com/ja-JP/policies/row-terms-of-use/>
<https://openai.com/api/pricing/>

■ Codex セキュリティ（一次情報）

<https://developers.openai.com/codex/security/>

■ Cerebras（一次情報）

<https://www.cerebras.ai/blog/openai-codexspark>

■ 競合（一次情報）

<https://www.anthropic.com/news/claude-opus-4-6>
<https://docs.anthropic.com/ja/docs/about-claude/pricing>
<https://ai.google.dev/gemini-api/docs/gemini-3>
<https://ai.google.dev/gemini-api/docs/pricing>
<https://cloud.google.com/vertex-ai/generative-ai/pricing>

■ 実運用の評判・議論（参考、一次情報ではない）

<https://community.openai.com/t/introducing-gpt-5-3-codex-spark-our-first-real-time-coding-model/1394497>
<https://github.com/openai/codex/issues/10953>
https://www.reddit.com/r/codex/comments/1r1ruqg/how_do_we_know_were_actually_getting_53_codex_and/
<https://news.ycombinator.com/item?id=43040784>

■ 報道（参考）

<https://techcrunch.com/2026/02/12/a-new-version-of-openais-codex-is-powered-by-a-new-dedicated-chip/>

1 5 17 60 <https://developers.openai.com/codex/security/>

<https://developers.openai.com/codex/security/>

2 3 4 7 8 9 11 12 13 14 23 24 25 28 35 53 54 55 59 65 <https://openai.com/index/introducing-gpt-5-3-codex-spark/>

<https://openai.com/index/introducing-gpt-5-3-codex-spark/>

6 10 36 37 56 57 67 68 <https://github.com/openai/codex/issues/10953>

<https://github.com/openai/codex/issues/10953>

15 34 <https://www.anthropic.com/engineering/infrastructure-noise>

<https://www.anthropic.com/engineering/infrastructure-noise>

16 <https://community.openai.com/t/gpt-5-3-codex-spark-research-preview-with-1000-tokens-per-second/1374091>

<https://community.openai.com/t/gpt-5-3-codex-spark-research-preview-with-1000-tokens-per-second/1374091>

18 22 38 66 <https://developers.openai.com/codex/changelog/>

<https://developers.openai.com/codex/changelog/>

19 <https://developers.openai.com/codex/pricing/>

<https://developers.openai.com/codex/pricing/>

20 52 <https://openai.com/ja-JP/policies/row-terms-of-use/>

<https://openai.com/ja-JP/policies/row-terms-of-use/>

21 <https://openai.com/business-data/>

<https://openai.com/business-data/>

26 69 <https://openai.com/index/cerebras-partnership/>

<https://openai.com/index/cerebras-partnership/>

27 <https://techcrunch.com/2026/02/12/a-new-version-of-openais-codex-is-powered-by-a-new-dedicated-chip/>

<https://techcrunch.com/2026/02/12/a-new-version-of-openais-codex-is-powered-by-a-new-dedicated-chip/>

29 30 41 <https://openai.com/index/introducing-gpt-5-3-codex/>

<https://openai.com/index/introducing-gpt-5-3-codex/>

31 32 33 61 <https://cdn.openai.com/pdf/23eca107-a9b1-4d2c-b156-7deb4fbc697c/GPT-5-3-Codex-System-Card-02.pdf>

<https://cdn.openai.com/pdf/23eca107-a9b1-4d2c-b156-7deb4fbc697c/GPT-5-3-Codex-System-Card-02.pdf>

39 43 45 <https://www.anthropic.com/claude/opus>

<https://www.anthropic.com/claude/opus>

40 44 46 <https://ai.google.dev/gemini-api/docs/gemini-3>

<https://ai.google.dev/gemini-api/docs/gemini-3>

42 50 <https://openai.com/api/pricing>

<https://openai.com/api/pricing>

47 51 <https://docs.anthropic.com/en/docs/about-claude/pricing>

<https://docs.anthropic.com/en/docs/about-claude/pricing>

48 <https://ai.google.dev/gemini-api/docs/pricing>

<https://ai.google.dev/gemini-api/docs/pricing>

49 <https://www.anthropic.com/news/claude-opus-4-6>

<https://www.anthropic.com/news/claude-opus-4-6>

58 [https://www.reddit.com/r/codex/comments/1r1ruqg/
how_do_we_know_were_actually_getting_53_codex_and/](https://www.reddit.com/r/codex/comments/1r1ruqg/how_do_we_know_were_actually_getting_53_codex_and/)

https://www.reddit.com/r/codex/comments/1r1ruqg/how_do_we_know_were_actually_getting_53_codex_and/

62 63 <https://openai.com/enterprise-privacy/>

<https://openai.com/enterprise-privacy/>

64 <https://openai.com/index/trusted-access-for-cyber/>

<https://openai.com/index/trusted-access-for-cyber/>