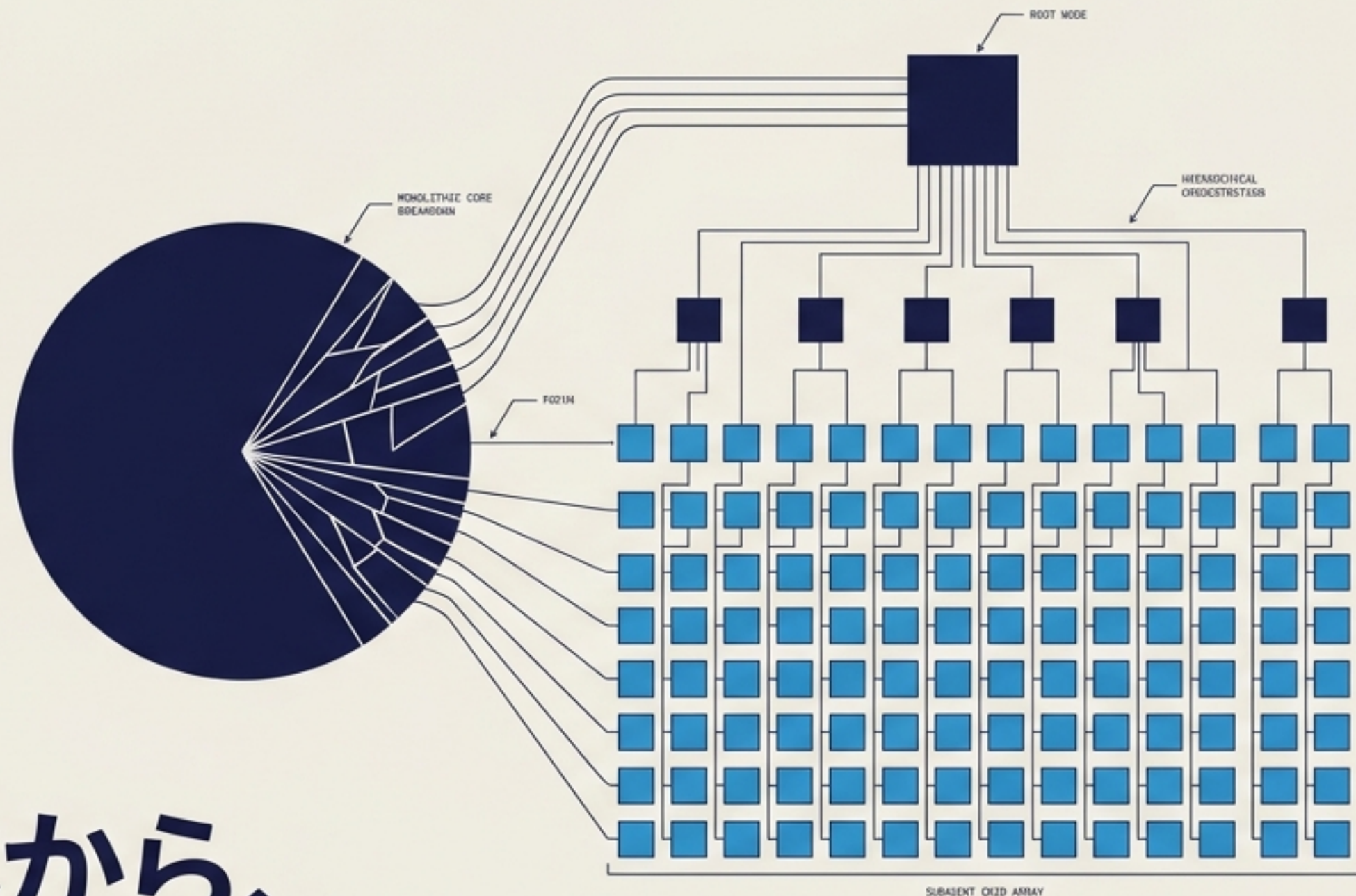




単体モデルの限界から、 協調アーキテクチャへ。

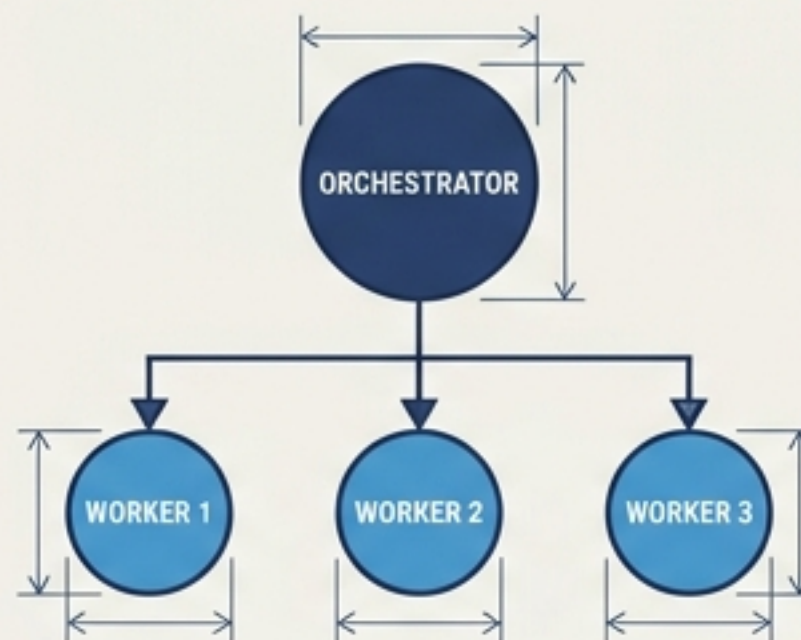
Claude Haiku 5.5がもたらすシステム設計のパラダイムシフト

単一モデルへの依存を脱却し、認知的階層化による並列実行基盤を構築する



推論力と圧倒的低コストの共存

Terminal-Bench等で前世代から劇的な性能向上（0.0%→39.2%）を果たしつつ、Effort Controlsにより計算量を動的に最適化する軽量ワーカーの誕生。



協調型アーキテクチャの台頭

巨大な単一モデルにすべてを委ねる設計から、Opus/Sonnetを司令塔とし、Haikuを並列ワーカーとして分散させる『認知的階層化』への移行。



コンテキストとリスクの厳格な制御

10万トークンを超える急激な価格高騰（5倍）や、小型モデル特有の過剰拒絶・視覚誤認を制御するエンジニアリングの必須化。

Claude Haiku 5.5：推論計算量を動的に制御する軽量モデルの到達点

コア・プロファイル

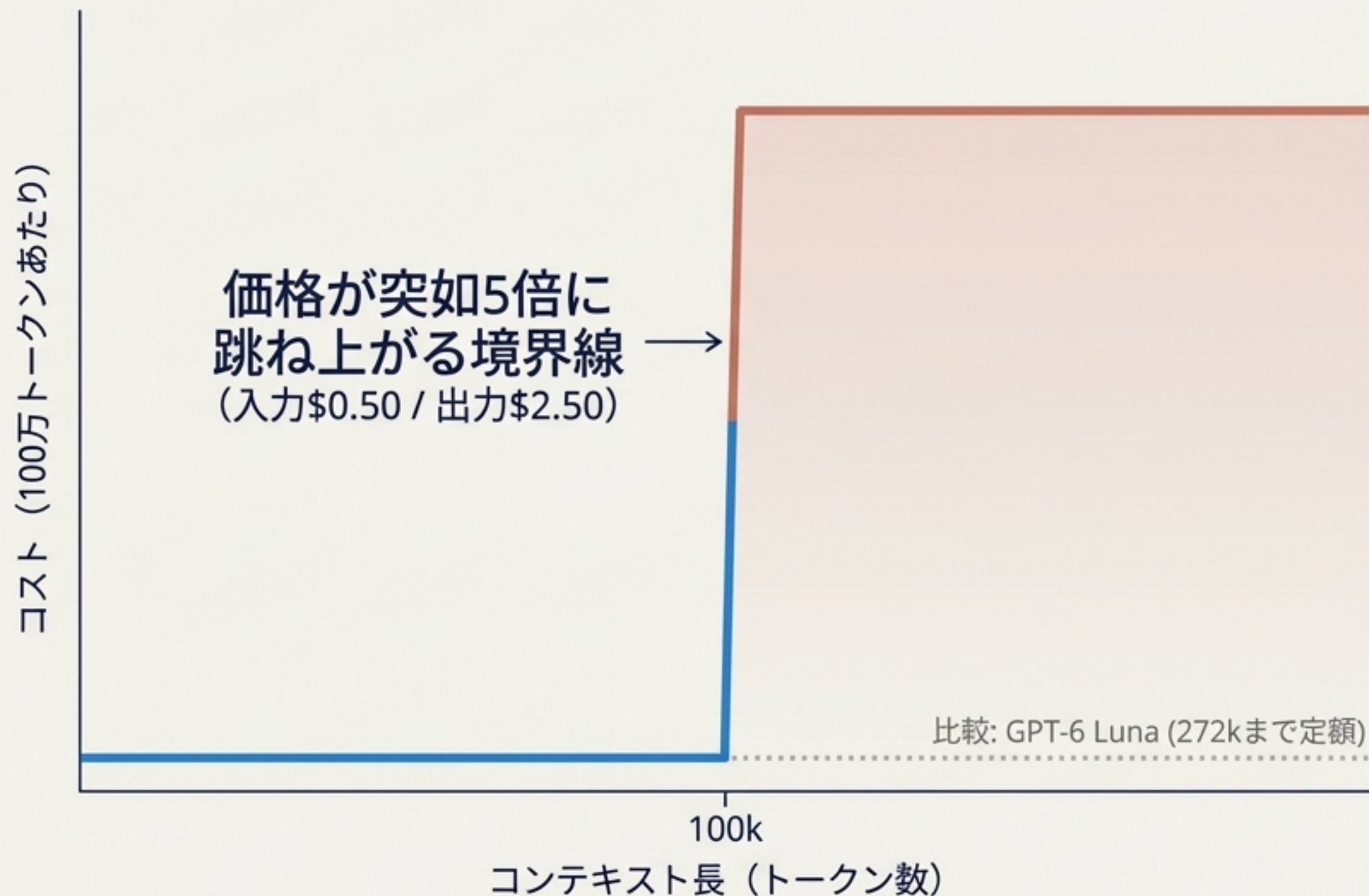
リリース日	2026年10月7日
知識カットオフ	2026年6月
入力モダリティ	高解像度マルチモーダル（画像・テキスト）
出力モダリティ	テキスト特化
トークナイザー	最新世代（前世代比消費量 1.25倍）



Effort Controls (Thinking Mode)

小型モデルの固定された知能から脱却。ワークロードの要件に応じ、推論品質とコストのバランスを動的に調整可能に。

『100,000トークンの崖』が示唆する、 局所的ワーカーとしての運用インセンティブ



見えない変数：トークナイザーの刷新

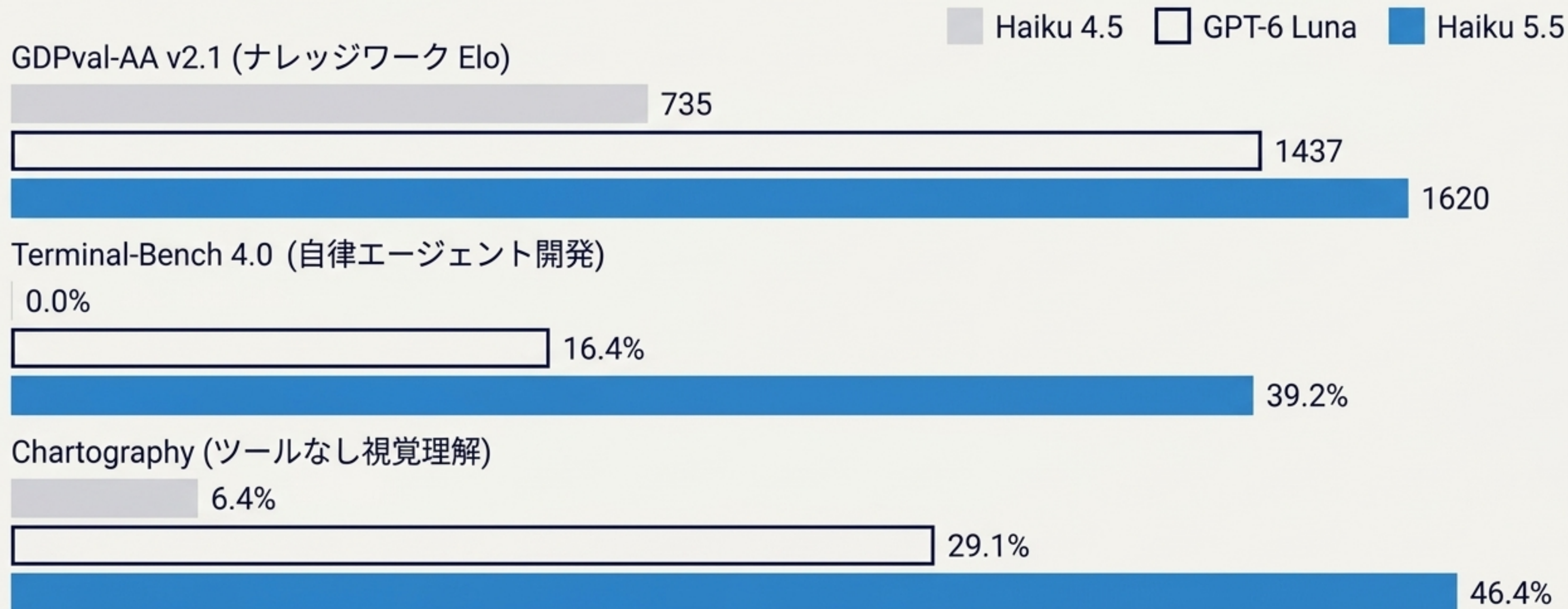
新トークン消費量 = 従来比 1.25倍

単価90%減 × 消費量1.25倍 = 実質コスト削減
約75%

※この非連続な価格構造は、Haiku 5.5を巨大な文脈に
用いるのではなく、局所的タスクに高頻度で呼び出す
よう強制するアーキテクチャ的メッセージである。

小型モデルの概念を覆す、上位モデルに迫るベンチマーク

特に自律エージェント環境において、前世代から根本的な能力刷新を果たし、競合を圧倒。



思考プロセスの拡張による文脈保持力の補償

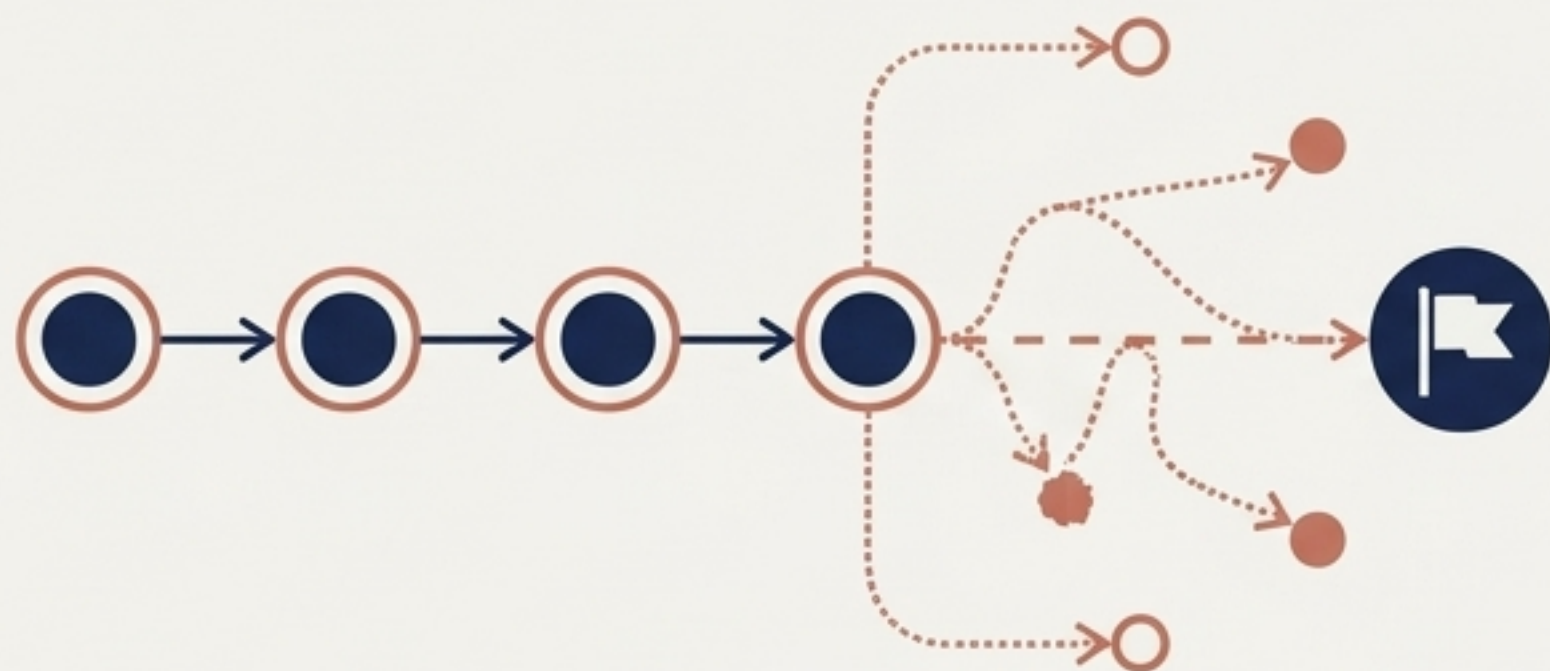
小型モデル特有の文脈保持力の限界を、思考ステップへの計算リソース割り当てによって補償する。
推論（Thinking）の有効化により、コード品質は上位モデル水準へと急伸する。

FrontierCode 1.1: Haiku 5.5 (Max)
46.4% vs GPT-6 Luna 42.4%



小型モデルの死角：長期的計画性の欠如と視覚のハルシネーション

状態遷移と長期計画の限界



↗ 部分達成率: 72.4% ↓ 厳密完全達成率: 37.1%

個々のUI操作は的確だが、目的地までの手順が伸びると整合性が崩れる。上位モデルのオーケストレーションが不可欠。

視覚抽出エラーと論理の暴走



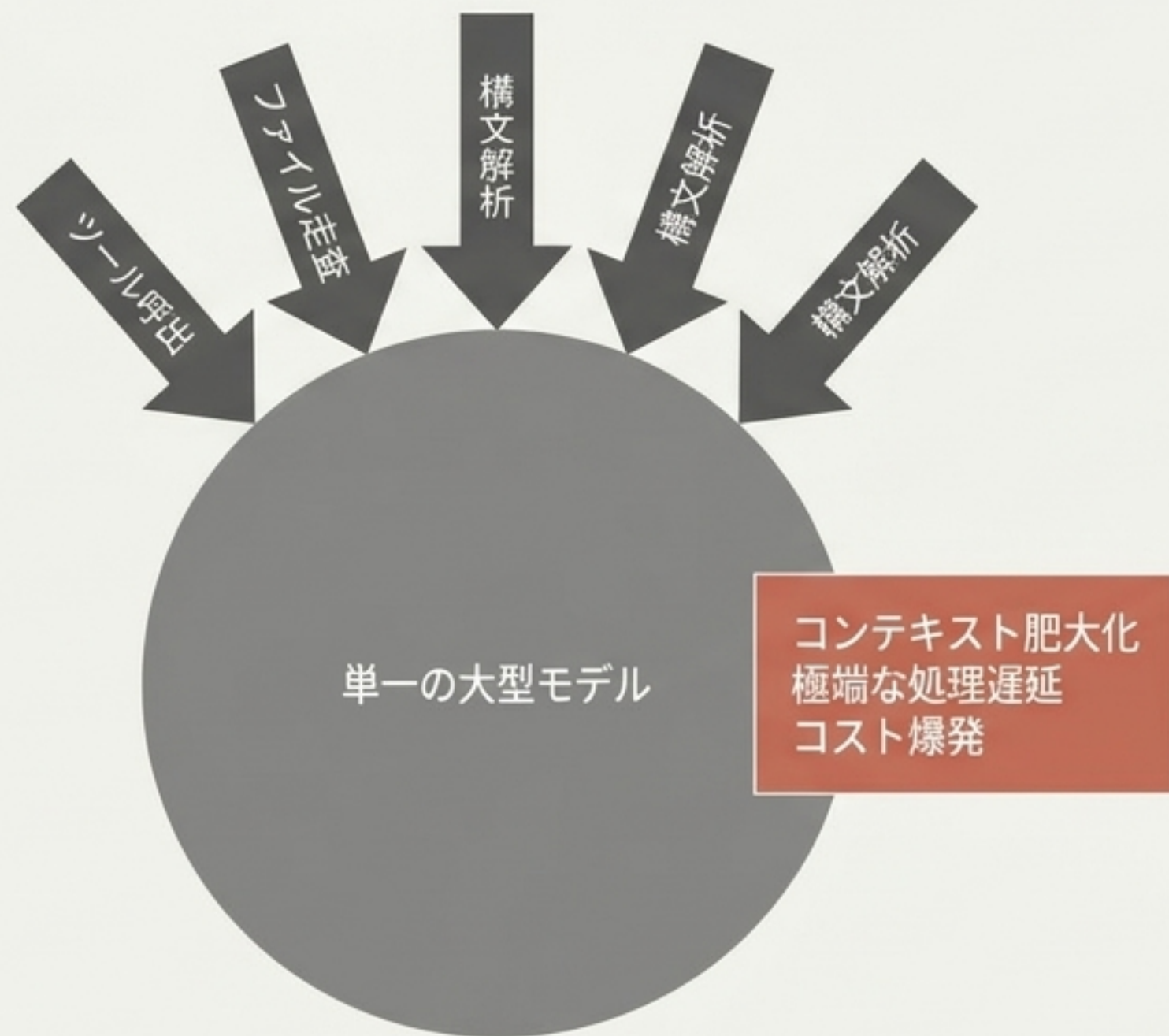
視覚抽出 (Visual Extraction) 論理推論 (Logical Reasoning) 行動計画 (Action Planning)

⚠ 視覚エラー率: 53.6%

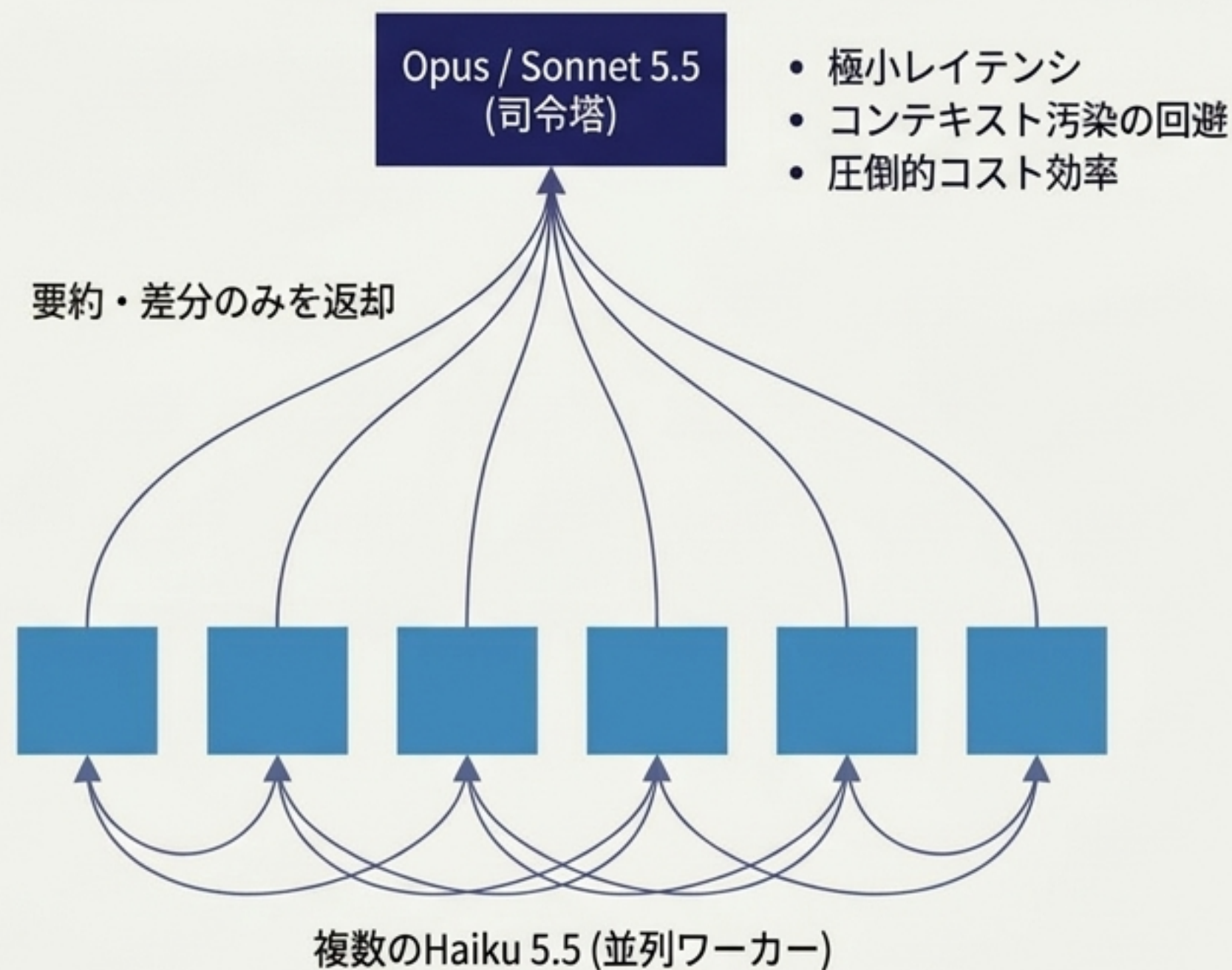
グラフ数値を視覚的に誤認した場合、その誤った前提に対して一切の破綻なく推論を展開してしまう。外部バリデーションが必須。

モノリシックなボトルネックからの解放。階層型協調アーキテクチャの誕生

BEFORE: モノリシック処理



AFTER: 非同期ファンアウト



認知的階層化マトリクス：司令塔とサブエージェントの最適な機能分担

階層 / モデル	主な責務・実行タスク	最適化指標
プランニング層 Claude Opus 5.5 / Sonnet 5.5	- 複雑なタスクの因果分解と推論戦略の立案 - 複数ワーカーの出力統合と最終レポート生成 - 難度の高い根本原因のデバッグ	高精度な推論力 大局的文脈理解
高速実行ワーカー層 Claude Haiku 5.5	- 特定ファイルの検索・Grep・差分生成 10-K等のピンポイント情報抽出 - 定型的なGUI操作、大量文書の事前スクリーニング - 反復的リファクタリング	低レイテンシ 極小コスト 高度な並列処理 (ファンアウト)

産業界における実証成果：協調アーキテクチャが叩き出すROI

Cognition (Devin Fusion)

Opus 5.5（リード）× Haiku 5.5（サイドキック）の協調アーキテクチャ。FrontierCodeでトップティアの66.2を維持しつつ、実行コストとレイテンシを大幅に削減。

Rogo

Opusが構成を統括する裏で、複数のHaiku 5.5が10-K（有価証券報告書）を並列走査。必要な売上高データをピンポイントで抽出し、高速な財務資料生成を実現。

Asana (AI Teammates)

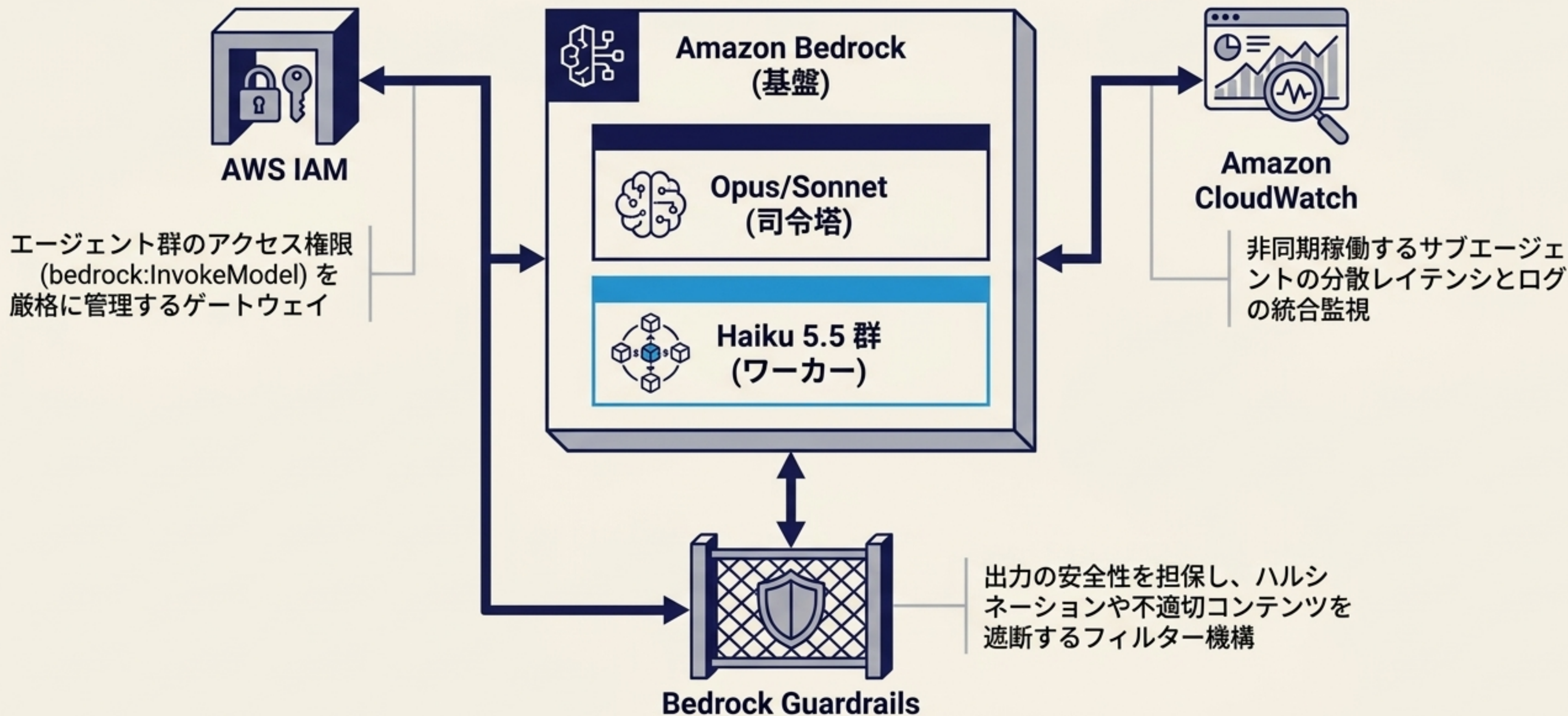
バグトリアージや大規模検索においてHaiku 5.5を採用。タスク完了レイテンシを30%削減し、推論速度を最大2.5倍に高速化。

HubSpot

CRM監査における休眠顧客の識別タスクに投入。全モデル中最速の完了速度を示し、3回試行平均で92.8%という最高精度を記録。

※その他、AlphaSense（正解率+0.08向上）、Box（遅延半減・スコア+11改善）などエンタープライズでの導入実績多数

AWS環境における複数エージェント統合のインフラストラクチャ設計



コンテキスト・エンジニアリング：コスト崩壊を防ぐパイプライン設計

アンチパターン（非推奨）



全データをそのまま流し込むと、100kの境界を越え、コストが即座に5倍に爆発する。

ベストプラクティス（推奨）



オーケストレーターが事前に数十～数百行の明確なスコープにチャンク化。全てのエージェントを100kの安全圏内で並列稼働させる。

サブエージェント特有のセキュリティ：堅牢性と過剰拒絶のトレードオフ

堅牢性の獲得



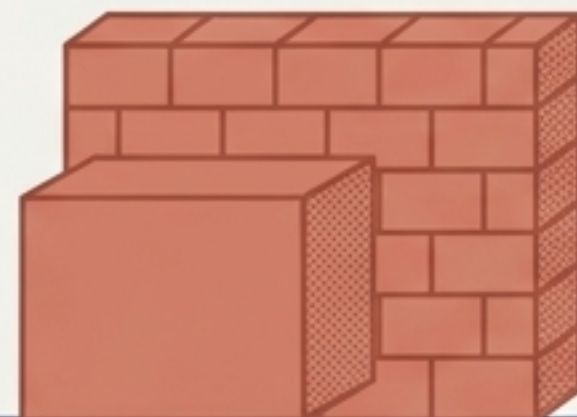
間接プロンプトインジェクション
攻撃の成功率が激減。

Haiku 4.5: 83.2%

Haiku 5.5: 7.1%

外部リソース（Web検索等）に直接
接触する最前線ワーカーとして、
十分な安全性を確立。

監査上のリスク



過剰拒絶（Over-Refusal）の増加：
無害な指示に対する防御機構の誤作
動。

漏洩コード解答の無断使用率：2.0%
から 17.0% へ上昇。

本番コードのライセンス管理におい
ては、人間または上位モデルによる
慎重な精査が必須。

結論：『どのモデルを選ぶか』から『いかに結線するか』への不可逆的転換



Claude Haiku 5.5の真の価値は、単体ベンチマークの向上にとどまらない。高価な思考力（Opus/Sonnet）と、極小レイテンシの実動ワーカー（Haiku）を有機的に統合する『認知的階層化』を経済的に成立させたことにある。AIシステム開発の競争軸は、もはや単一モデルの選定ではなく、『並列実行基盤のインテグレーション能力』へと完全に移行した。

The Modular Blueprint: Key Takeaways

1

Haiku 5.5の再定義

単なる定型タスク処理器から、Effort Controlsを備え、実環境を自律操作できる『実動ワーカー』への決定的な進化。

2

経済的制約のハック

10万トークンの価格境界（崖）と消費トークン膨張を逆手にとり、強制的に並列ファンアウト設計を促すアーキテクチャ戦略。

3

堅牢な実行層の確立

インジェクション耐性7.1%という、汚染リスクのある外部データ走査に耐えうる最前線のセキュリティ獲得。外部検証との組み合わせが鍵となる。



Sources:
- Anthropic Claude Haiku 5.5 System Card
- AWS Machine Learning Blog
- Cursor Docs & OSWorld Benchmarks