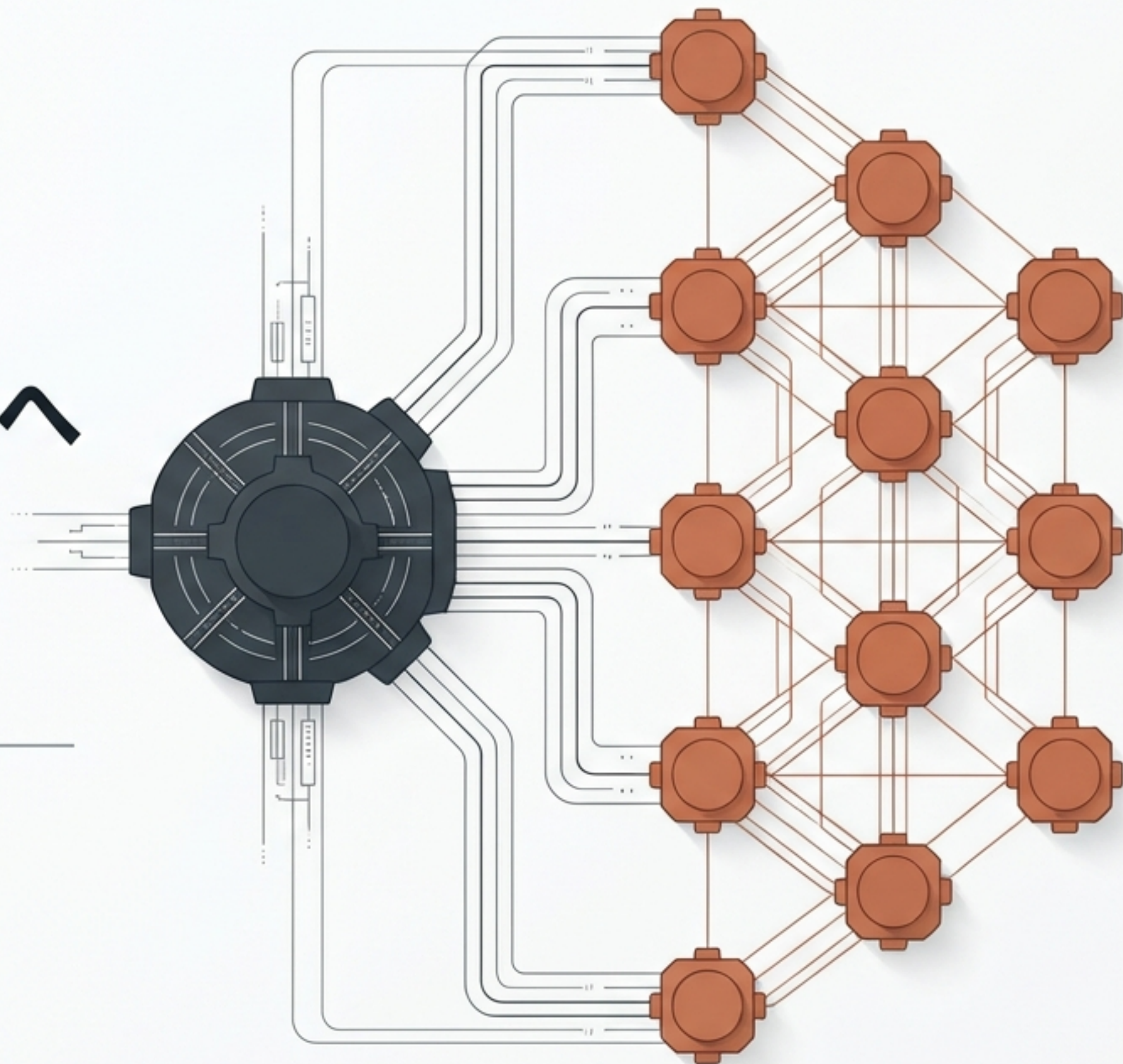


小型モデルの評価軸は 「単体性能」から 「サブエージェント適性」へ

Claude Haiku 5.5 調査レポート：
実務における真価と組み込み戦略

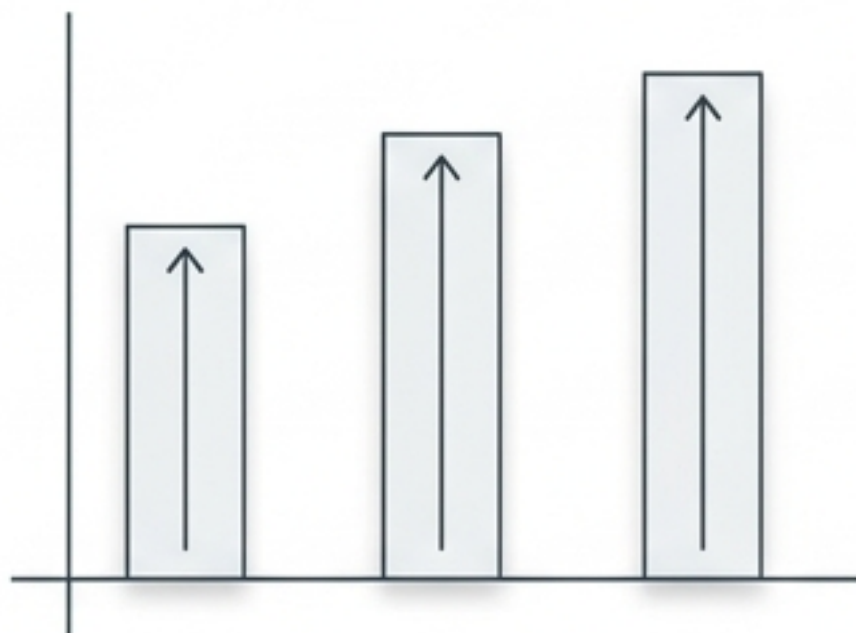
2026年10月8日

対象：AI戦略立案者・リードエンジニア・DX担当者



単体スコア競争の終焉と、チームとしての貢献度評価への移行

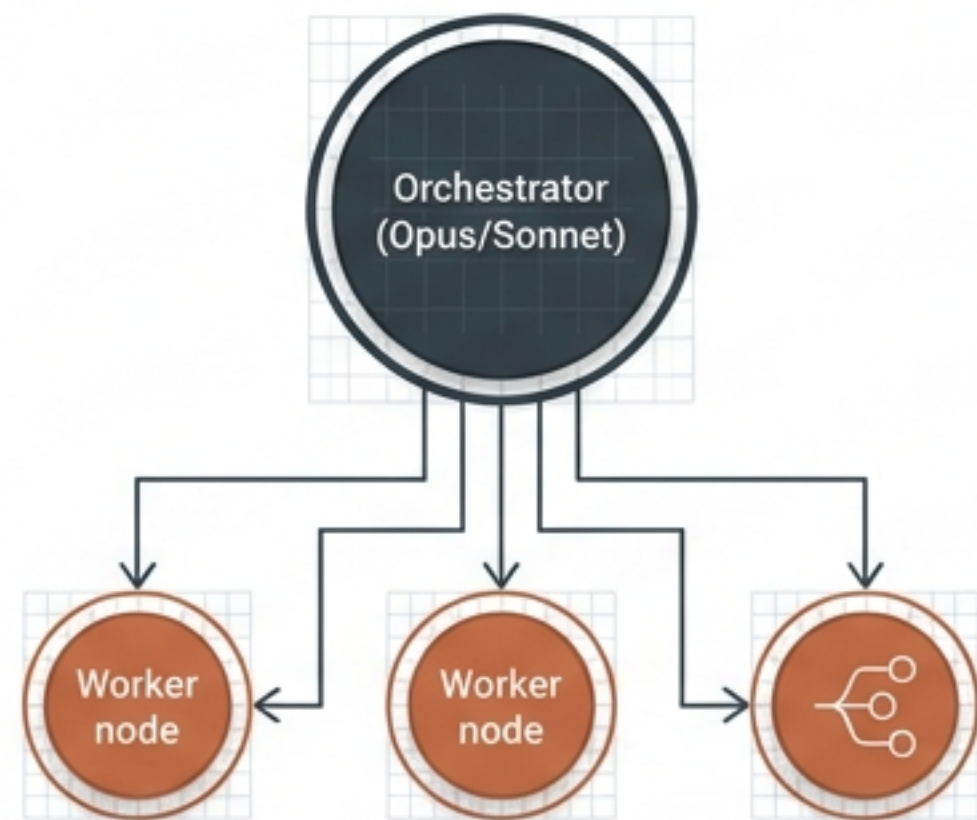
Before



単体性能の追求

「このモデルは賢いか？」という単一の評価基準。

After (The Paradigm Shift)



サブエージェント適性

「上位モデル（Opus/Sonnet）の指揮下で、いかに安く、速く、確実に動けるか？」へのシフト。単体比較が不要になのではなく、新たな評価軸が「重ね合わせ」られた状態。

Claude Haiku 5.5 の全容：並列処理に特化したスペック



Scale

100万トークン

前世代（20万）からの大幅拡張。最大出力も12.8万トークンへ。



Cost

90%コストダウン

10万トークン以下の入力で\$0.10。出力は\$0.50（100万トークンあたり）。

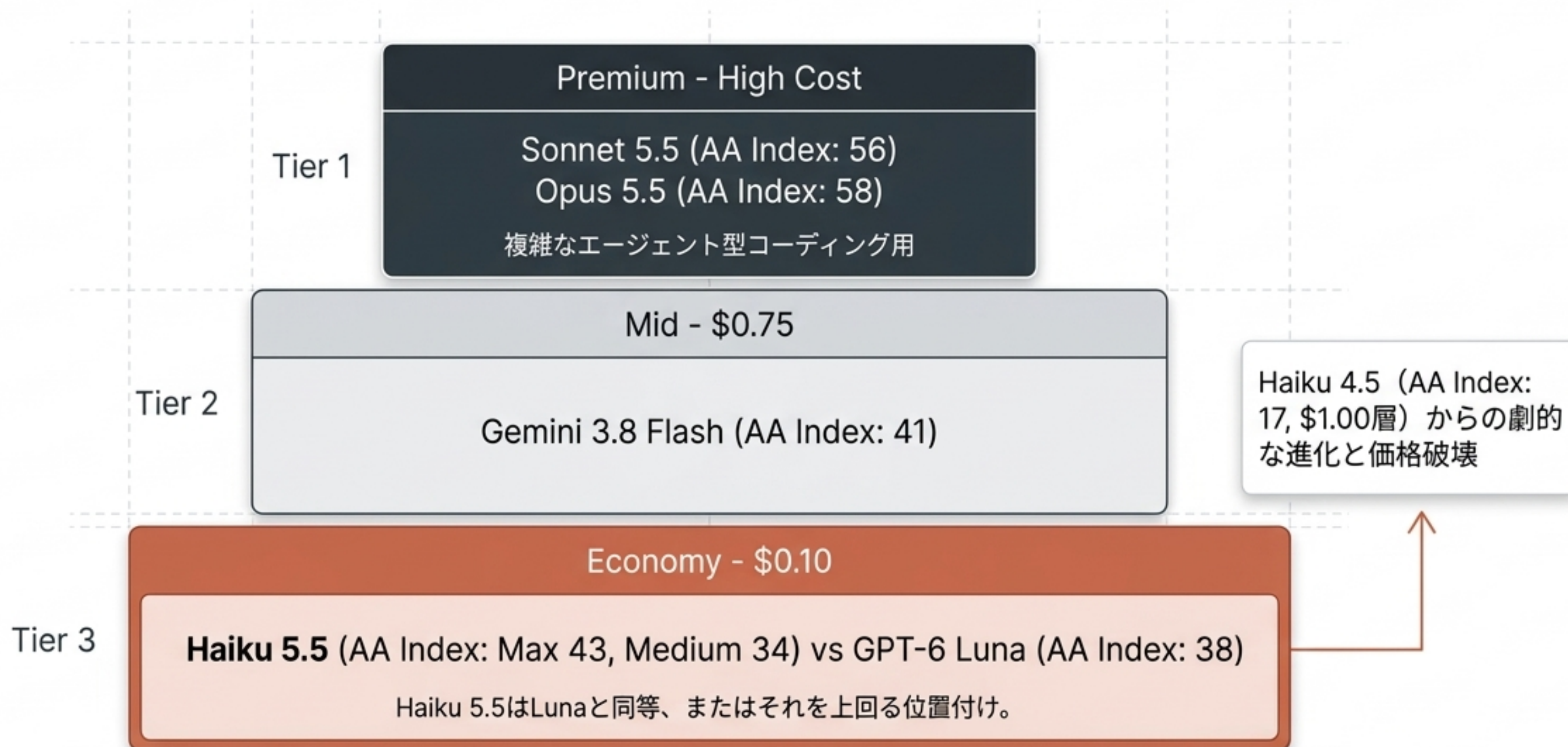


Engine

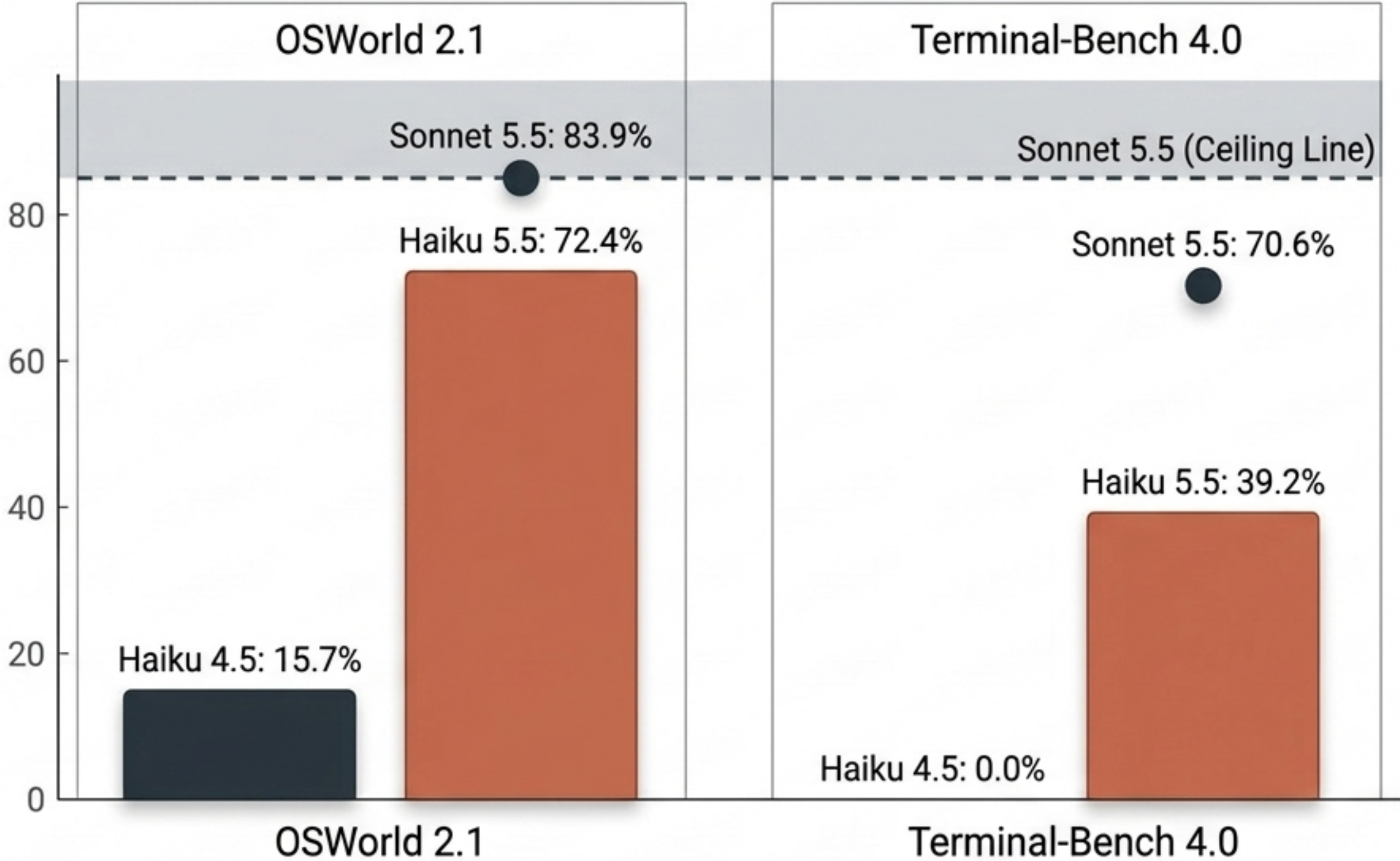
Adaptive thinking

推論機能がデフォルトでON。タスクに応じて「low」から「max」まで5段階の労力 (effort) 調整が可能。

同価格帯（\$0.10層）において最上位クラスのパフォーマンスを獲得



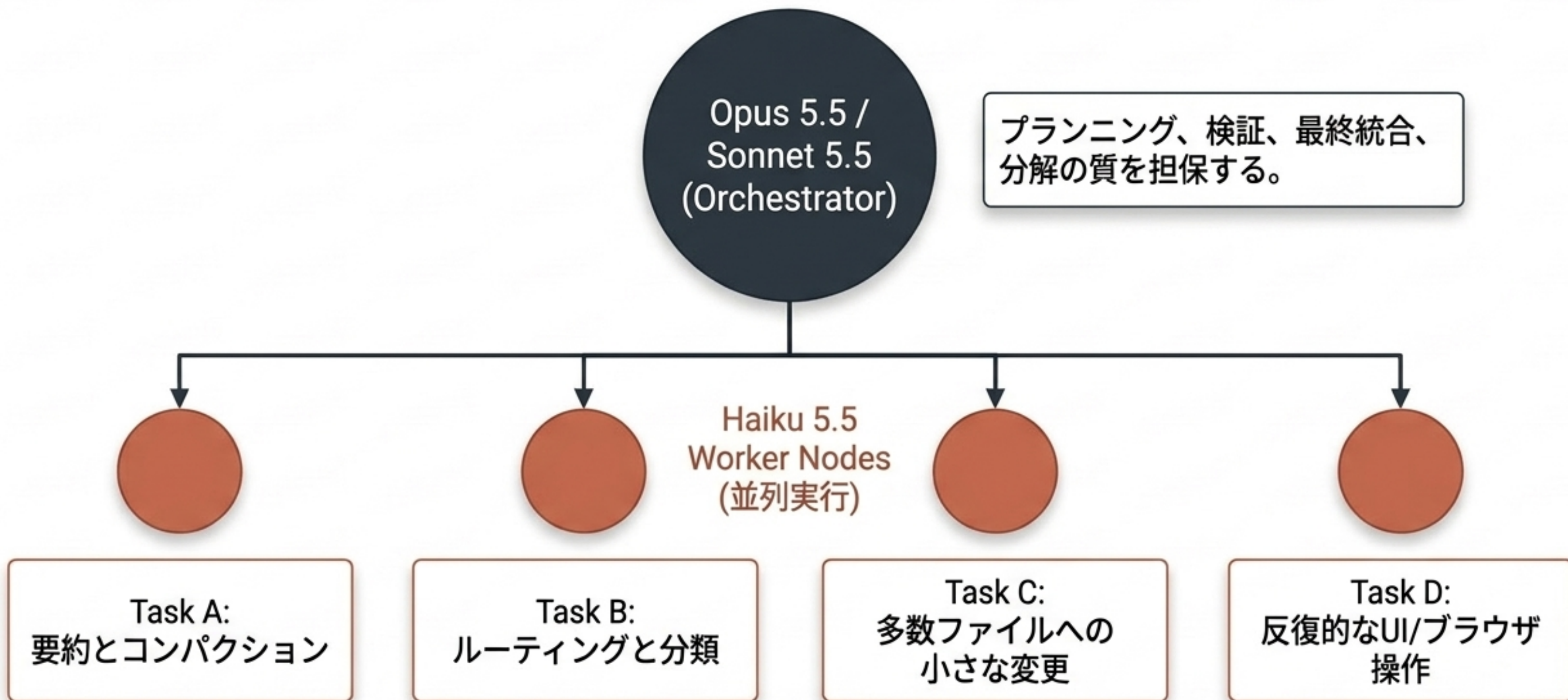
意図的に設けられた「単体性能の限界」



UI Key Insight Panel

Anthropic自身が「複雑なタスクには Sonnet/Opusを使うべき」と明言。単体で最強を目指すのではなく、用途を限定したポジション。

最適解は「オーケストレーター」と「ワーカー」の組み合わせ



実運用における「システムとしての性能」が 単体スコアを凌駕する

COGNITION / DEVIN FUSION

FrontierCode 66.2達成

Haiku単体（46.4）を大きく上回る。Opus 5.5をリード、**Haiku 5.5**を補佐役とする構成でコストとレイテンシを削減。

ASANA

完了レイテンシ 30%短縮

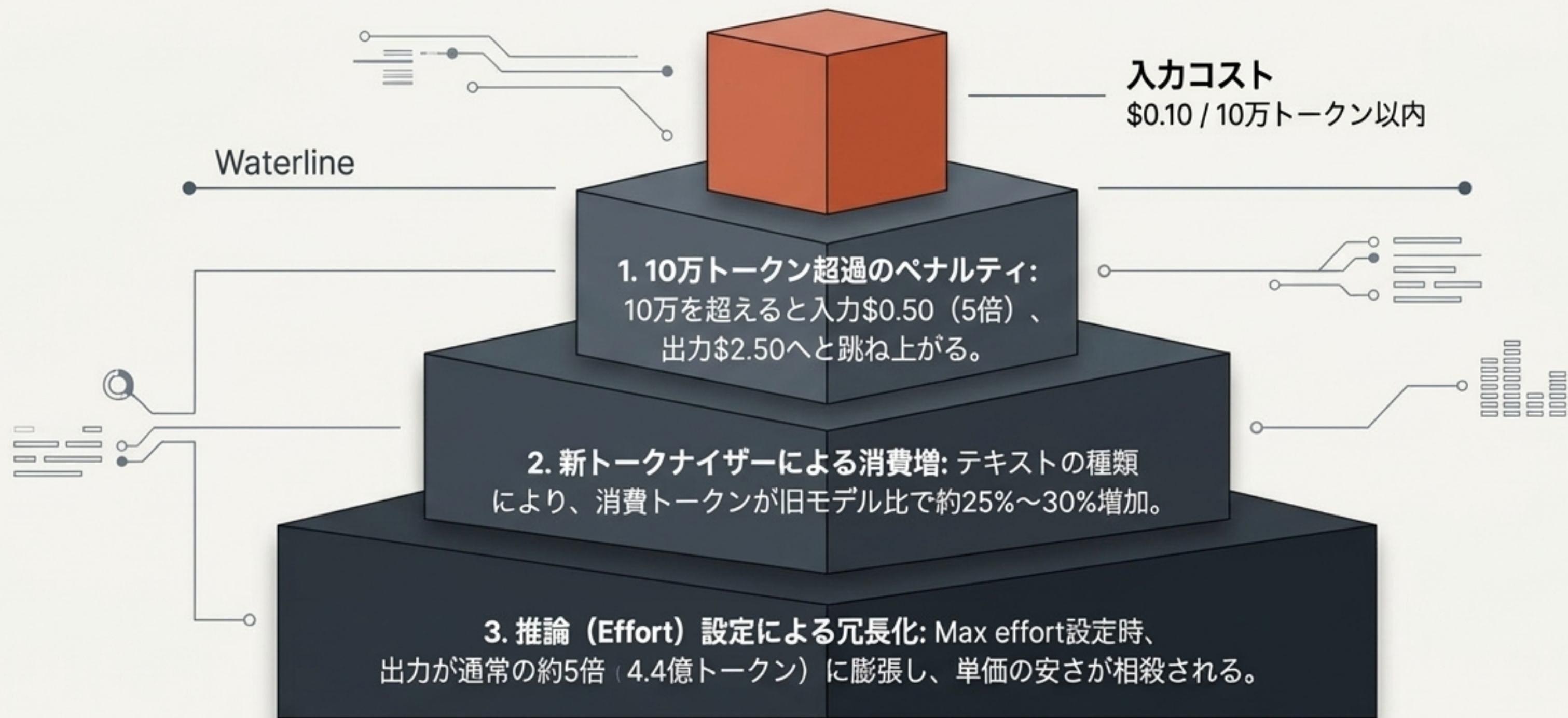
AI Teammatesの評価にて、エージェント1ターンあたり最大2.5倍の高速化。

ALPHASENSE

精度の大幅 向上

週800万コールの文書クエリで、0.76（**Haiku 4.5**）から0.84へと統計的に有意な改善。

カタログ上の安さ（\$0.10）の裏に潜む「隠れコスト」



堅牢なセキュリティの代償となる「過剰拒否」のリスク

プロンプトインジェクション耐性の劇的向上



間接攻撃の成功率が83.2% (4.5) から7.1%へと激減。悪意あるUI操作の拒否率もOpus/Sonnetを上回る。

過剰拒否 (Over-refusal) の頻発



行動監査において「テストした全モデル中で最も過剰拒否が多い」と報告。安全分類器によるエラーからのフォールバックが存在しないため、業務停止のリスクあり。

開発環境（Claude Code）への組み込み時に陥りやすい罠

CLI/Terminal

THE TRAP: DEFAULT BEHAVIOR

```
$ claude-code
```

> Running Explore/Plan sub-agent...

> Using inherited parent model: claude-3-opus-20240229

THE SOLUTION: OPTIMIZED CONFIGURATION

```
$ CLAUDE_CODE_SUBAGENT_MODEL_FORCE=claude-3-haiku-20240307 claude-code
```

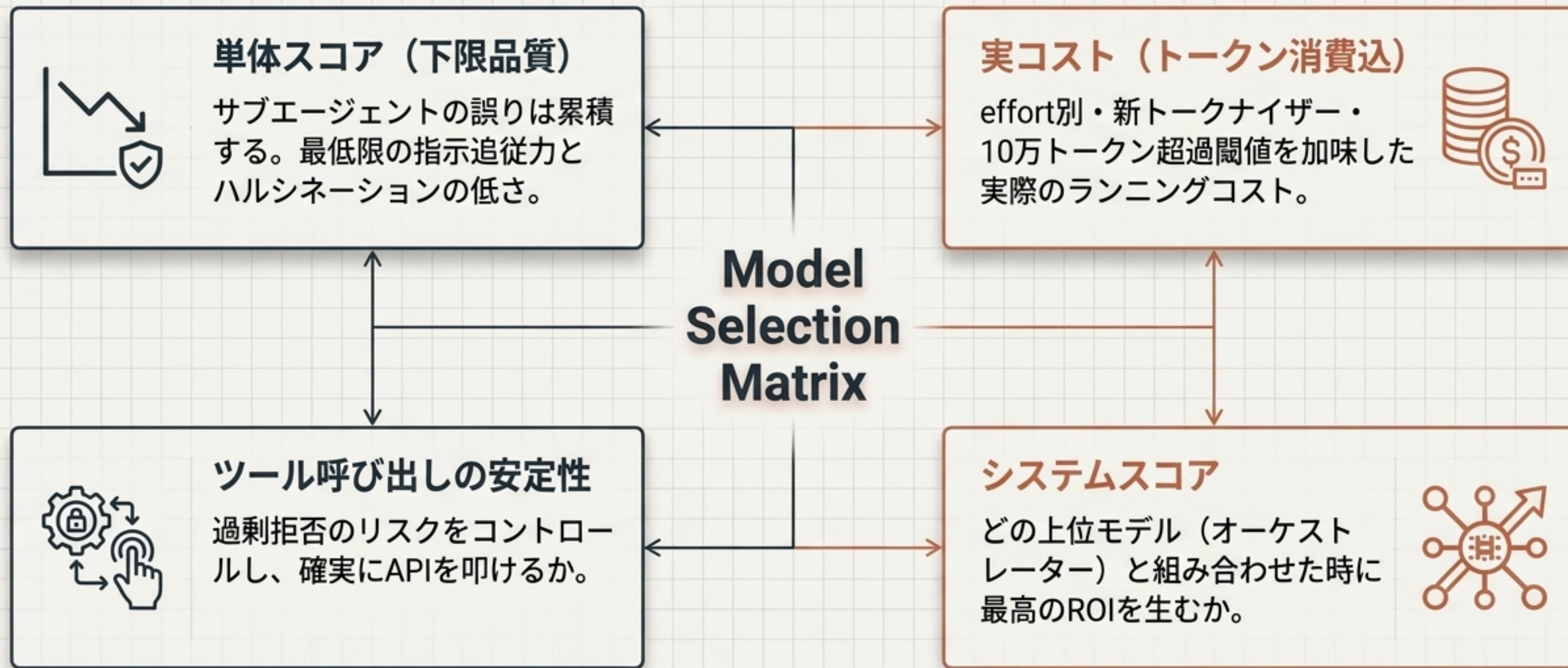
> Running Explore/Plan sub-agent...

> Using explicitly configured model: claude-3-haiku-20240307 (Haiku 5.5)

デフォルトのExplore/Planサブエージェントは「親のモデルを継承」するため、指定しない限り安価なHaiku 5.5では動かない。

コストを最適化するためには、独自エージェントに「model: haiku」を明示的に指定するか、環境変数の設定が必須。

新時代における小型モデル選定の「4つの評価軸」



段階的な導入とシステム単位でのA/Bテストが最適解となる

01

Scope (要件定義)

medium effortかつ入力が10万トークン以内に確実に収まるサブタスク（分類・要約）に限定して導入検証を開始する。

02

Contrast (単体vs連携の比較)

「Haiku 5.5単体構成」と「Opus 5.5 + Haiku 5.5構成」を同一のタスクで比較。



Haiku 5.5
単体



Opus 5.5 +
Haiku 5.5 連携

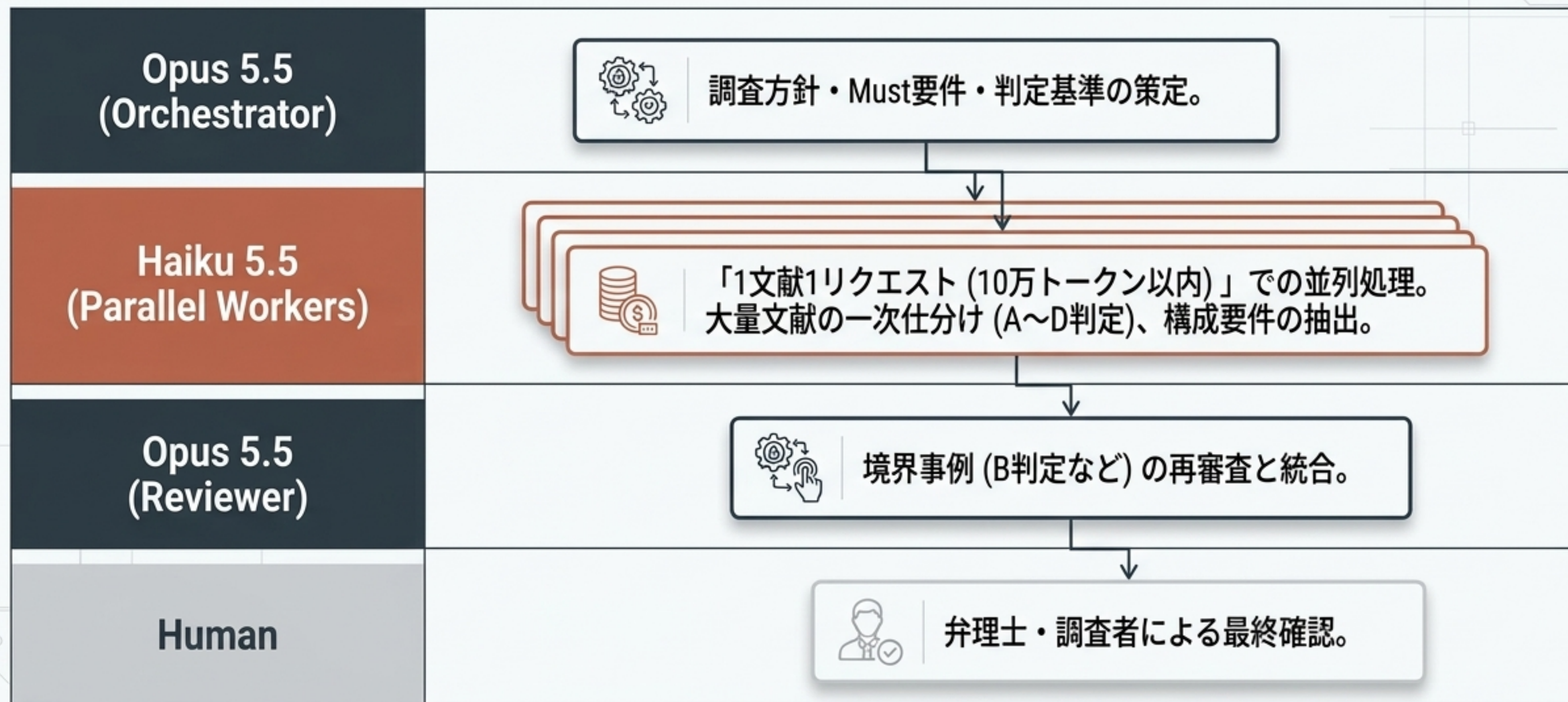
03

Measure (システム指標での評価)

単純な正答率だけでなく「全体のコスト」「処理の所要時間」「上位モデルによるやり直し率」を総合的に測定する。



実務応用例：知財業務におけるマルチエージェント・ワークフロー



専門業務（知財実務）への導入におけるフェイルセーフ設計



✓ 誤判定前提の二重チェック

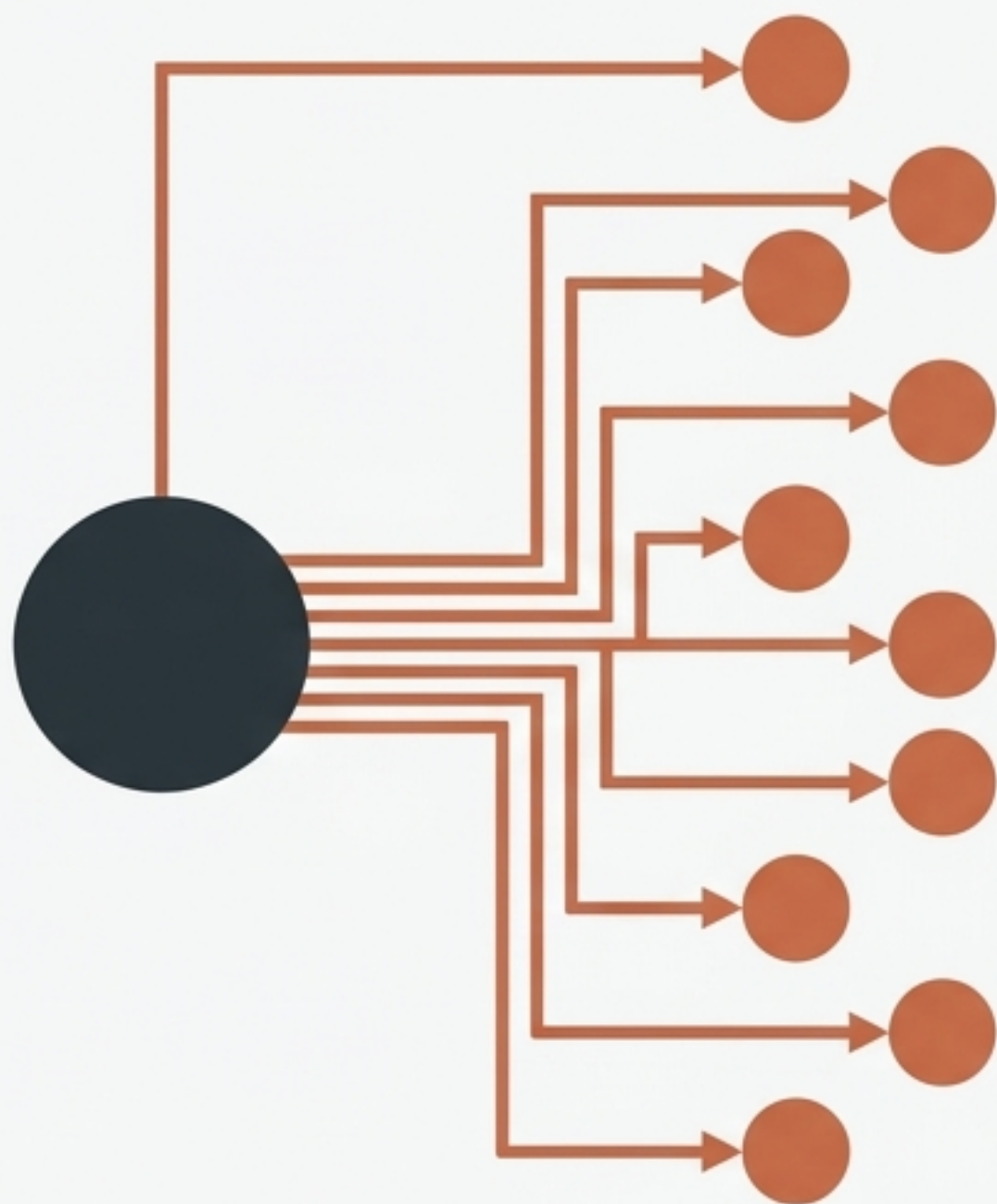
生成AIの特許スクリーニングにおける誤判定率（約2割）を前提に、見逃しが致命的な無効資料調査等では人間との連携を必須とする。

✓ 過剰拒否の事前テスト

Haiku 5.5特有の過剰な安全判定が、専門用語や特許文献の解析をブロックしないか実データで検証する。

✓ 機密性の担保

出願前情報の処理には、Amazon Bedrock等のリージョン内処理やエンタープライズ向けのデータ取り扱い条件を適用する。



次代の競争力は 「単体モデルの選定」から 「アンサンブルの設計」へ

Claude Haiku 5.5は単なる安価なモデルではありません。上位モデルの手足として機能する高度な並列処理システムの一部です。カタログスペック上の比較を終え、オーケストレーターとワーカーによる『チーム』をどう設計し、実装するかが、これからのAI導入における真の優位性となります。