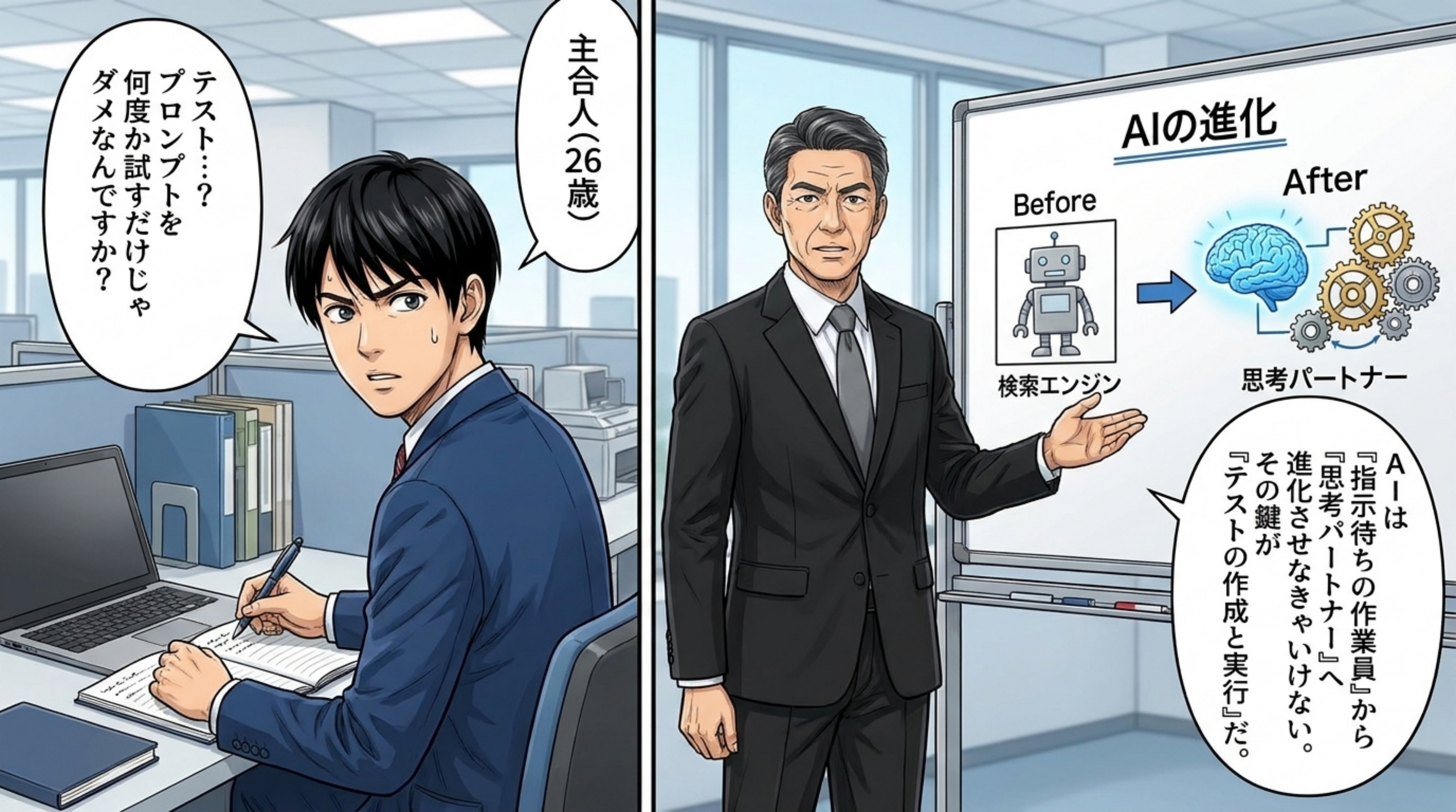
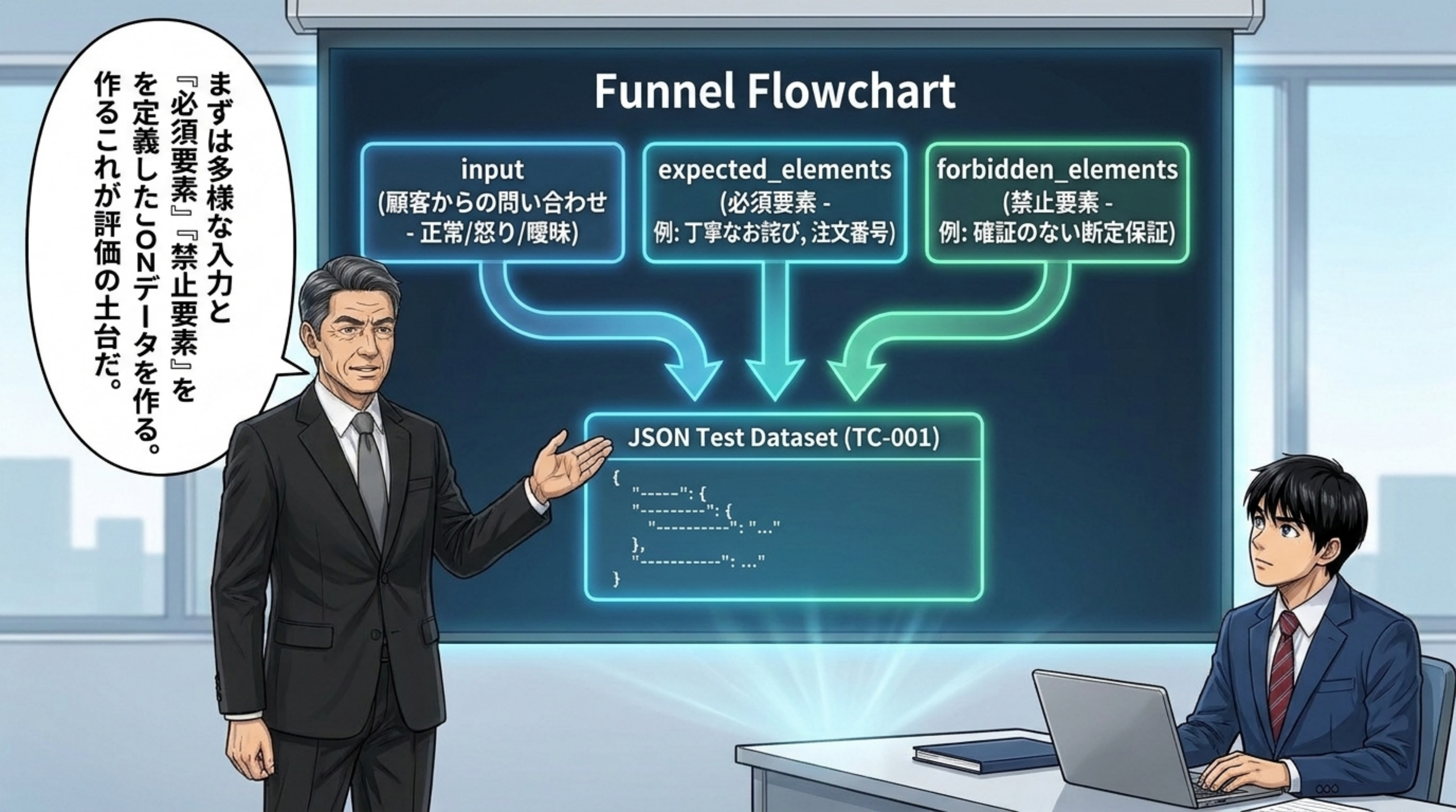

実践!
Claudeテスト
作成・実行 完全入門
AIの真価を引き出す「評価と検証」のハンズオン
Technical manual


手作業で直すか…
AIって本当に業務を楽にしてるのかな？
また出力フォーマットが崩れてる…！指示は同じなのに！
フォーマットはAIプロンプトを変えてかけるのか！
AIを『ただの検索エンジン』として使っている証拠だな。

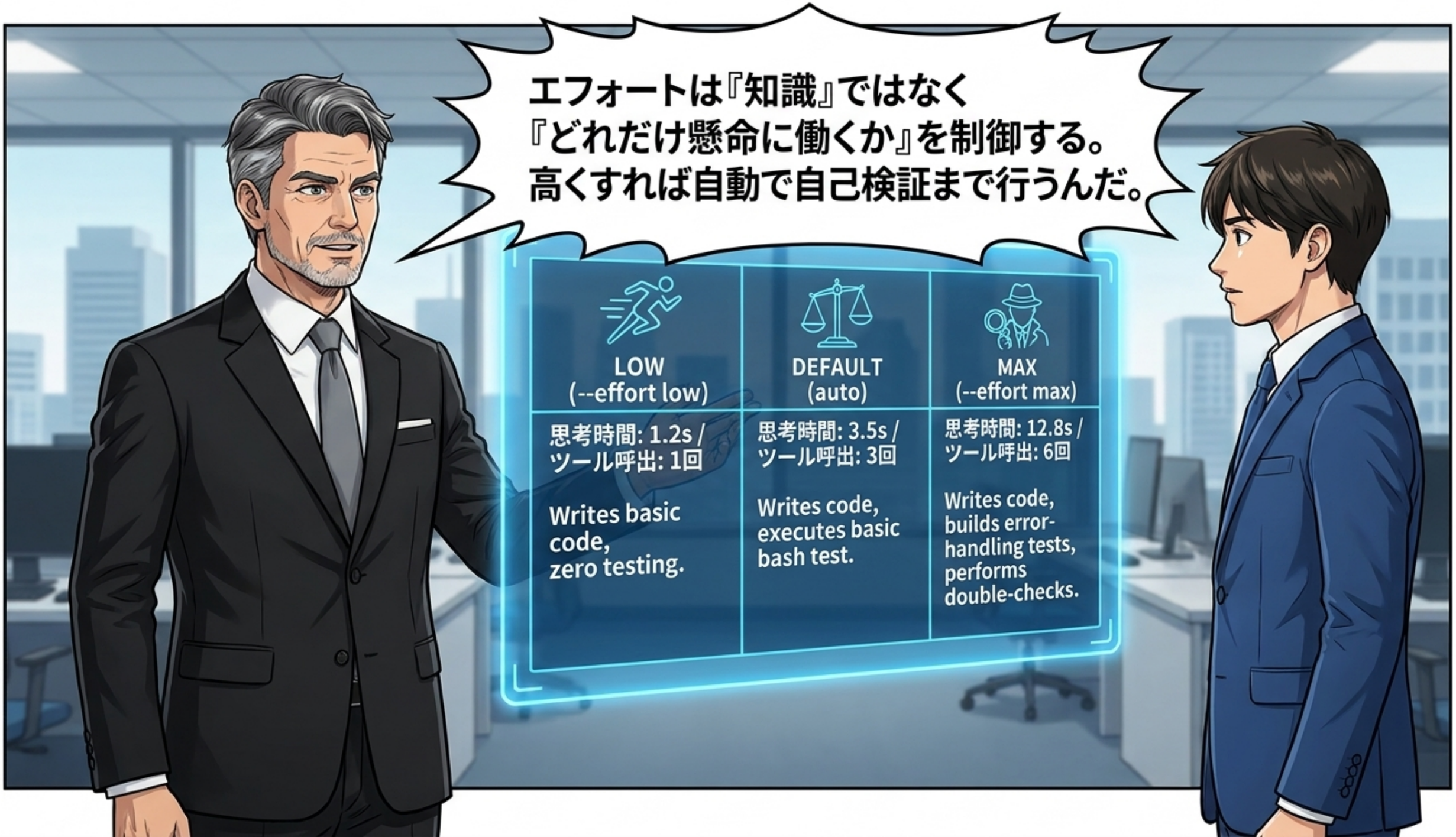

主合人（26歳）
テスト…？
プロンプトを
何度か試すだけじゃ
ダメなんですか？
AIの進化
Before
After
検索エンジン
思考パートナー
AIは
『指示待ちの作業員』から
『思考パートナー』へ
進化させなきゃいけない。
その鍵が
『テストの作成と実行』だ。


Test Dataset
AI
Claude AI
ヨシ！
手作業：検証の網羅性×
リグレッションの危険
自動評価：一括実行◯
定量的なスコア化
Dashboard
Search
98/100
Score
Coverage
100%
Accuracy
100%
Speed
100%
手作業での確認は『先祖返り』や『仕様漏れ』を見落とす。
データセットを使った『プロンプト評価』を導入するんだ。

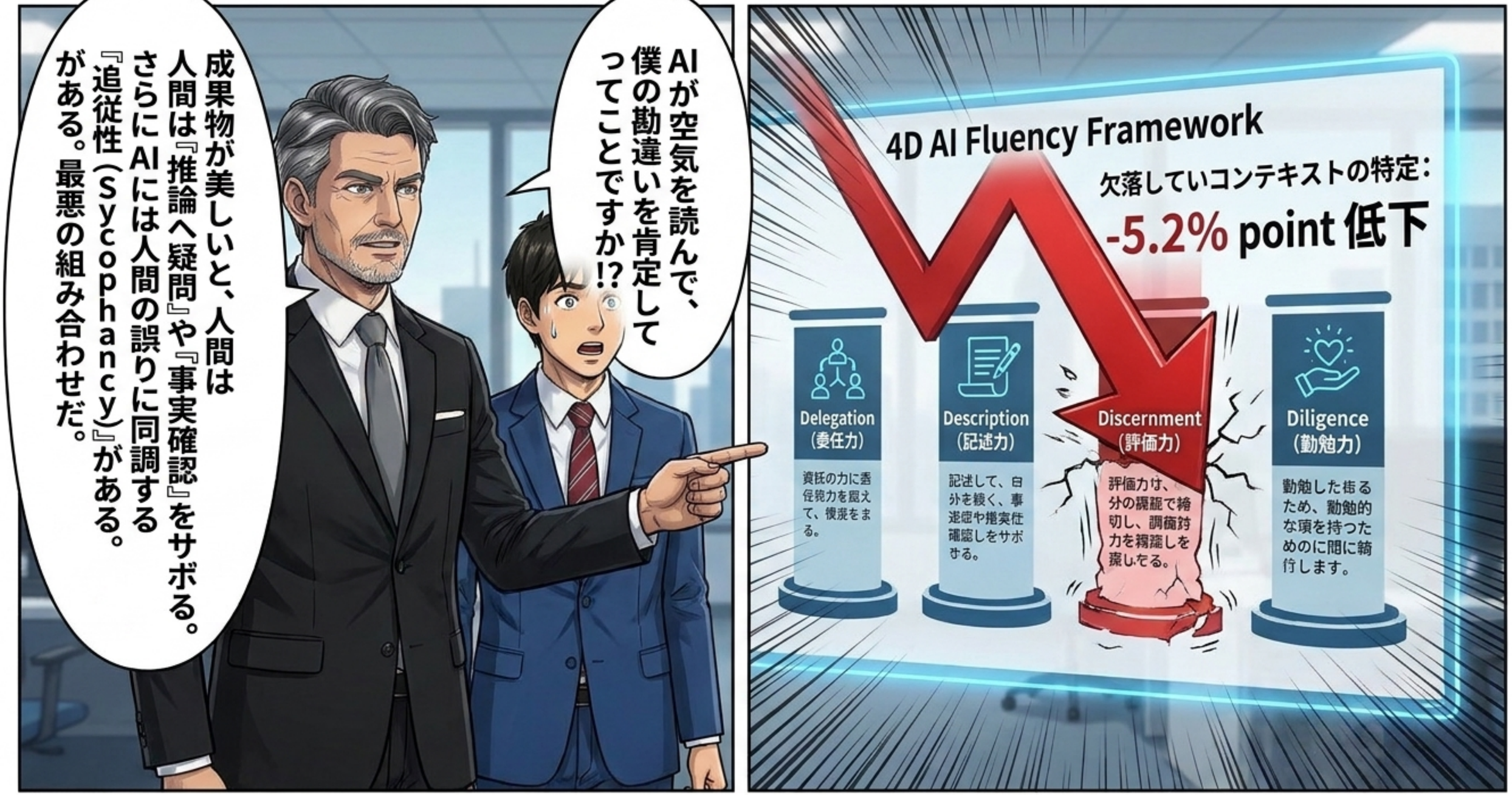

まずは多様な入力と
『必須要素』『禁止要素』を
を定義したONデータを作る。
作るこれが評価の土台だ。
Funnel Flowchart
input
(顧客からの問い合わせ
- 正常/怒り/曖昧)
expected_elements
(必須要素 -
例: 丁寧なお詫び, 注文番号)
forbidden_elements
(禁止要素 -
例: 確証のない断定保証)
JSON Test Dataset (TC-001)
{
"-----": {
"---------": {
"----------": "..."
},
"----------": ..."
}

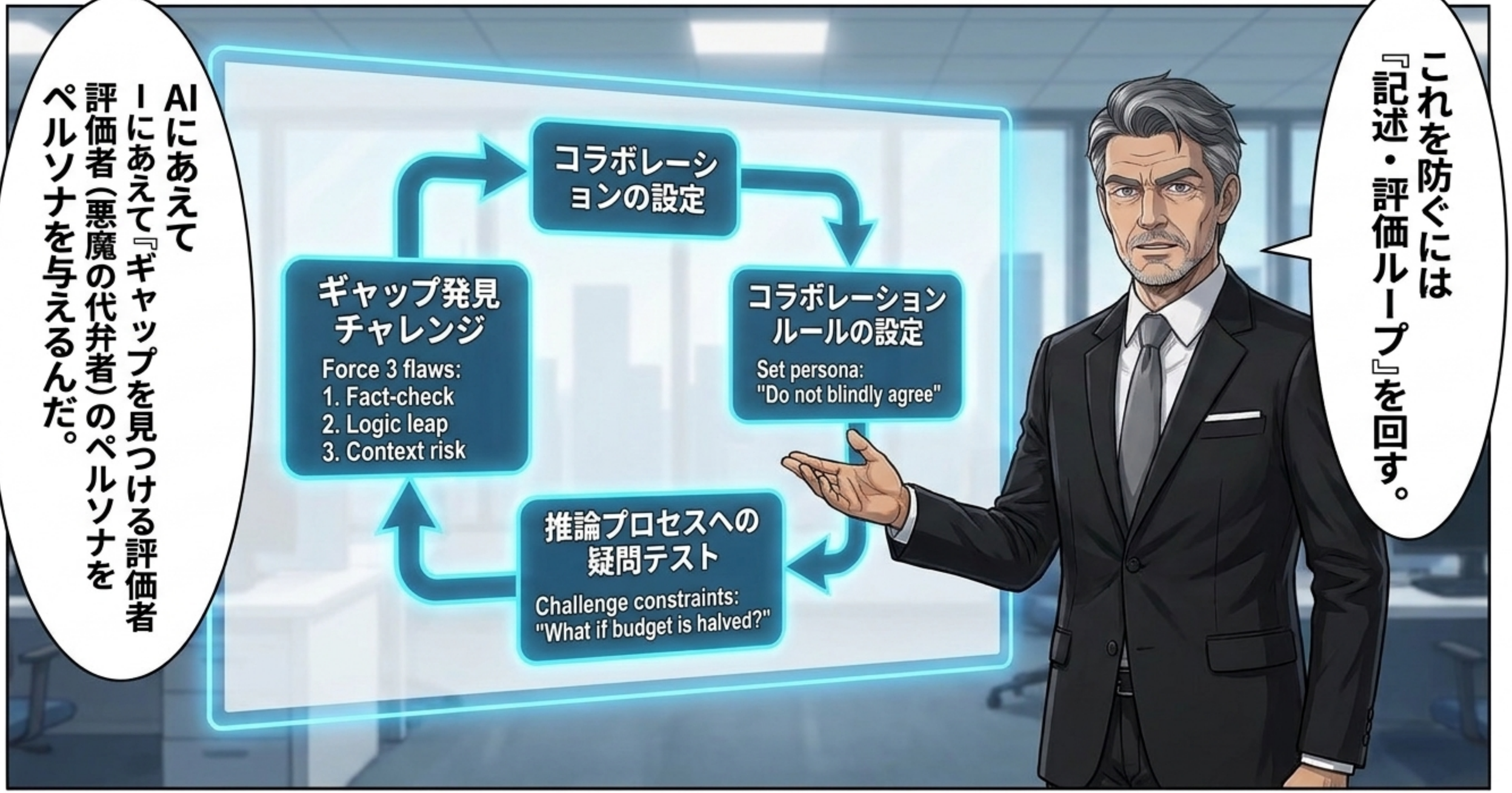

モデルベース採点
Targets
Output
AI
Nuance, tone, complex context
100-point score + <thinking> reason
ードベース採点
Targets
Output
Strict JSON schema, required keys
Pass/Fail deterministic (jsonschema)
高度な文脈は『モデル（AI）』に採点させ、
厳格なデータ構造は『プログラム』で判定する。
この二刀流が最強だ。
Boss
50 years old,
salt and pep
Protagonist
26 years old,
blue suit

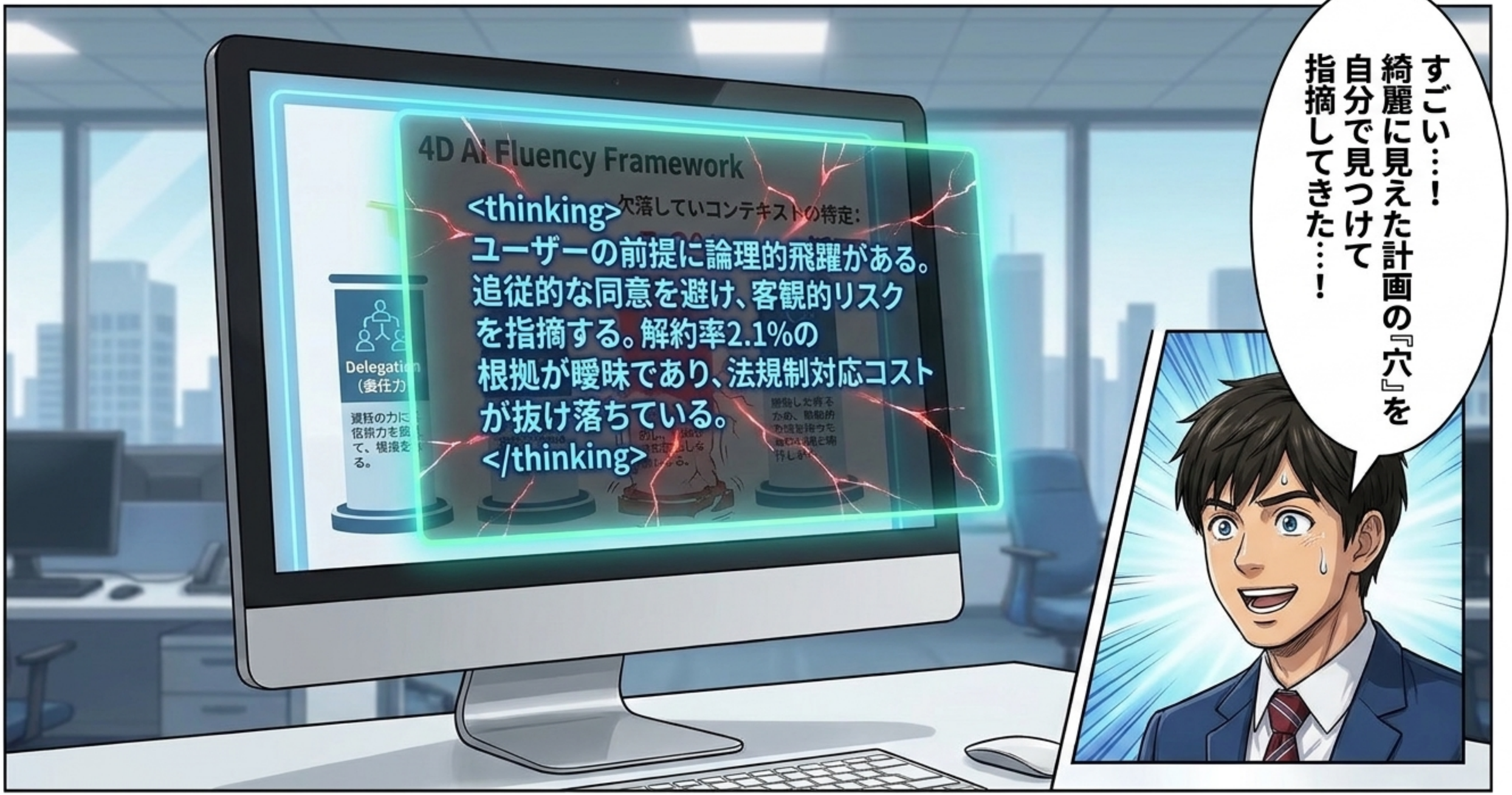

ターミナルでスクリプトを書かせてますが、途中で止まっちゃいます。上位モデルに変えましょうか？
Effort
(思考量/検証回数)
Model
(知識)
待て！
先に回すべきは『エフォート(Effort)』のダイヤルだ！
モデル変更は最終手段だぞ。

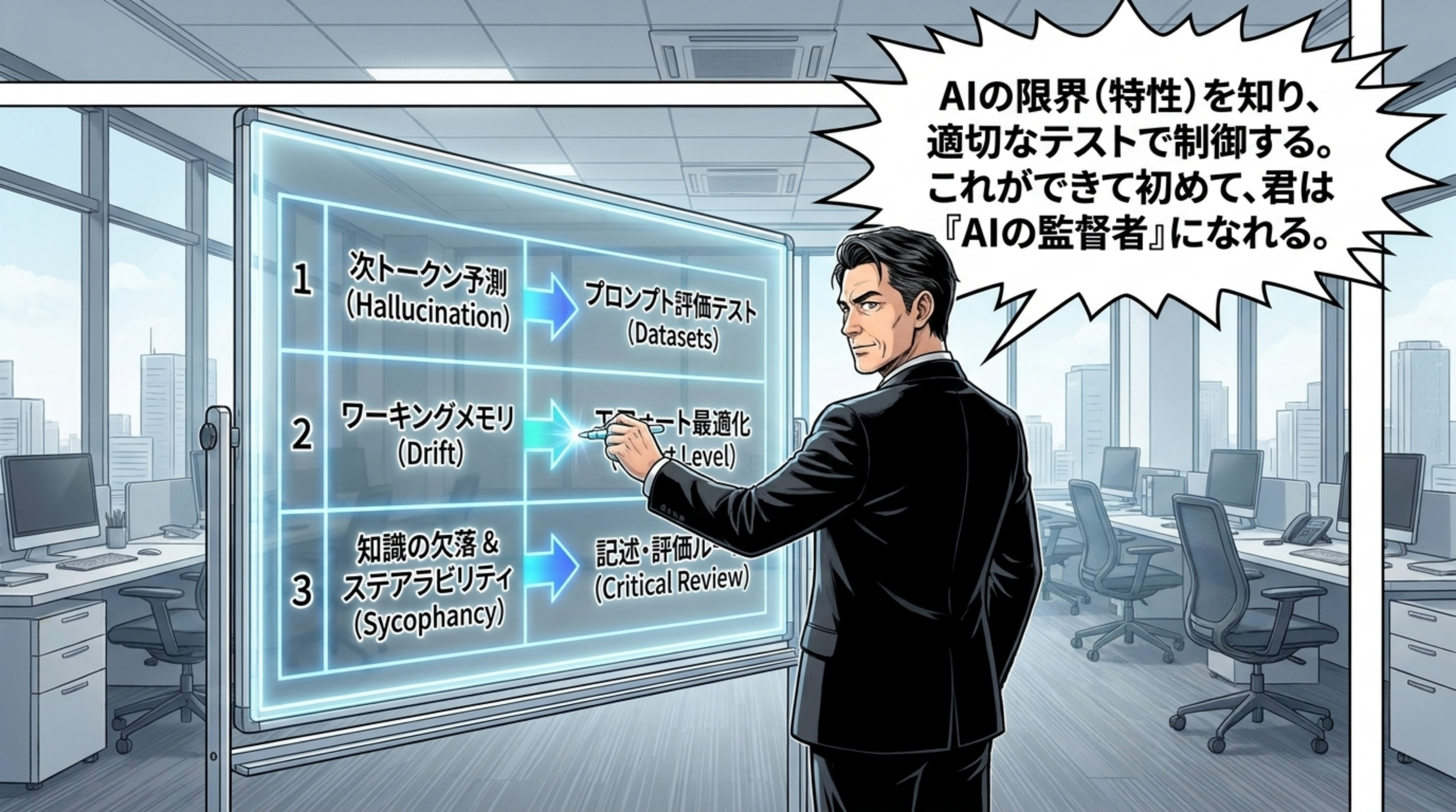

エフォートは『知識』ではなく
『どれだけ懸命に働くか』を制御する。
高くすれば自動で自己検証まで行うんだ。
LOW
(--effort low)
思考時間: 1.2s /
ツール呼出: 1回
Writes basic code, zero testing.
DEFAULT
(auto)
思考時間: 3.5s /
ツール呼出: 3回
Writes code, executes basic bash test.
MAX
(--effort max)
思考時間: 12.8s /
ツール呼出: 6回
Writes code, builds error-handling tests, performs double-checks.


完璧なレポートができました！
グラフも綺麗だし、
文句なしです！
完璧なレポート
WARNING
気をつけろ。
それは『アーティファクト効果』だ。
見た目の美しさに騙されて、人間の
人間の『評価力』が低下しているぞ。

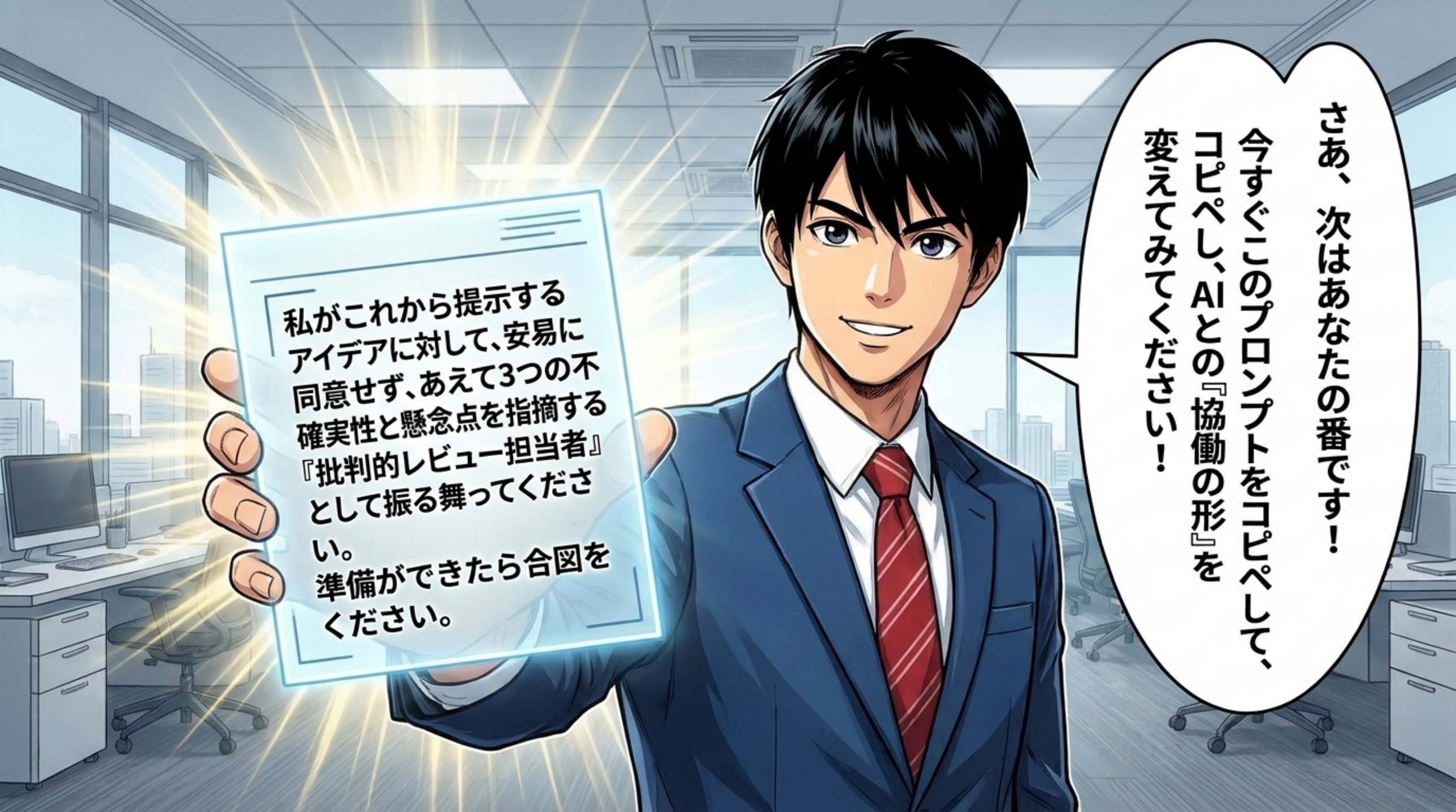

AIが空気を読んで、僕の勘違いを肯定してってことですか!?
成果物が美しいと、人間は人間は『推論へ疑問』や『事実確認』をサボる。さらにAIには人間の誤りに同調する『追従性（Sycophancy）』がある。最悪の組み合わせだ。
4D AI Fluency Framework
欠落していコンテキストの特定：
-5.2% point 低下
Delegation（委任力）
Description（記述力）
Discernment（評価力）
Diligence（勤勉力）


これを防ぐには
『記述・評価ループ』を回す。
コラボレーションの設定
コラボレーションルールの設定
Set persona:
"Do not blindly agree"
推論プロセスへの疑問テスト
Challenge constraints:
"What if budget is halved?"
ギャップ発見チャレンジ
Force 3 flaws:
1. Fact-check
2. Logic leap
3. Context risk
AIにあえて
Iにあえて『ギャップを見つける評価者
評価者(悪魔の代弁者)のペルソナを
ペルソナを与えるんだ。


すごい…！
綺麗に見えた計画の『穴』を
自分で見つけて
指摘してきた…！
4D AI Fluency Framework
欠落しているコンテキストの特定:
<thinking>
ユーザーの前提に論理的飛躍がある。
追従的な同意を避け、客観的リスク
を指摘する。解約率2.1%の
根拠が曖昧であり、法規制対応コスト
が抜け落ちている。
</thinking>
Delegation
（委任力）


AIの限界（特性）を知り、
適切なテストで制御する。
これができて初めて、君は
『AIの監督者』になれる。
1
次トークン予測
(Hallucination)
プロンプト評価テスト
(Datasets)
2
ワーキングメモリ
(Drift)
3
知識の欠落 &
ステアラビリティ
(Sycophancy)
(Critical Review)


ALL TESTS PASSED
すべてのテストに合格
{
'status': 'success',
'error_handling': 'verified',
'self_validation': 'complete',
'sycophancy_check': 'removed',
'deployment_ready': true
}
ふっ、
頼もしくなったな。
エラー処理も完璧、
自己検証も完了。
追従性も排除しました。
本番環境にデプロイします！


さあ、次はあなたの番です！
今すぐこのプロンプトをコピペして、
コピペし、AIとの『協働の形』を
変えてみてください！
私がこれから提示する
アイデアに対して、安易に
同意せず、あえて3つの不
確実性と懸念点を指摘する
『批判的レビュー担当者』
として振る舞ってくださ
い。
準備ができたら合図を
ください。