

知財業務におけるAIエージェント適用のための ハーネス設計とガバナンス・アーキテクチャ

Gemini 4 Argon

1. エグゼクティブ・サマリーとハーネス優位性仮説の知財的意義

大規模言語モデル(LLM)を知的財産業務(先行技術調査、明細書およびクレームのドラフティング、拒絶理由通知への応答、FTO調査等)へ適用する際、実務上の成否を分ける最大の要因はベースモデルのパラメータ規模ではなく、モデルを包摂するソフトウェア基盤である「エージェント・ハーネス(Agent Harness)」の設計にある¹。ベースとなるLLMは本質的に確率的な推論エンジン(Probabilistic Reasoning Engine)に過ぎず、いかに高度な推論能力を備えていても、単体では特許法上の厳格な様式要件、実体的要件、および弁理士法や各国特許庁が課す善管注意義務を担保することはできない²。確率的な推論を、再現性と法的整合性を備えた決定論的なアクションへと変換する制御機構こそがハーネスである¹。

近年のシステム工学およびAIアーキテクチャ研究(Meng, 2026等)では、エージェントは「モデル＋ハーネス」として定義され、サイバネティクス制御器(Cybernetic Governor)としてのハーネス H は以下の6要素のタプルとして形式化されている¹。

$$H = (E, T, C, S, L, V)$$

ここで、 E は実行環境およびサンドボックス隔離(Environment)、 T はツール定義・レジストリおよびディスパッチ機構(Tools)、 C はトークン予算下でのコンテキスト編成・圧縮(Context)、 S はセッション間およびステップ間の状態永続化(State)、 L は制御ループおよびライフサイクル管理(Loop / Lifecycle)、そして V は出力スキーマ検証・安全性チェック・停止条件を含む検証ゲート(Verification)を指す²。実装アーキテクチャにおいては、エージェントの静的ロジック(使用するモデルやプロンプトの青写真)を定義する「フレームワーク」、並行処理・耐障害性・サンドボックス制限下でプロセスを実行し能力の上限を物理的に制約するインフラ層の「ランタイム」、そしてモデルにツールやコンテキストを供給して思考・行動をオーケストレーションし評価・検証を行うアプリケーション層の「ハーネス」を明確に分離して設計する必要がある⁶。

実証データは、モデルのウェイト(重み)を固定したままハーネスのスキヤフォールディング(再試行ポリシー、テストフィードバック、コンテキスト圧縮手法)を変更するだけで、SWE-bench等の実務的タスク解決率が20パーセントポイント以上変動することを示している(拘束条件仮説: Binding Constraint Thesis)²。また、先行研究(Voyager等)では、再利用可能な手続き型コード抽象化としての永続的スキルライブラリをハーネスに組み込むことで、標準的なメモリ検索と比較して技術マイルストーンの発見速度が15.3倍に加速することも実証されている²。一方で、2023年頃の初期的な自律型エージェン

トで多用された単純な while(true) 型のReActループ(思考と観察を無制限に生テキストとしてコンテキストへ追記し続ける方式)は、トークンコストを10倍から1000倍に膨張させるだけでなく、単一ファイルのベンチマークから複数ファイルの実運用環境へ移行した際に20%から27%もの性能低下(Verified-to-Pro Gap)や推論の迷走を招くことが判明している²。現代のエンタープライズ知財システムにおいては、状態を持つ有向非巡回グラフ(StateGraphs / DAG)と動的リデューサー関数、条件付きエッジルーティング、およびチェックポイント化されたHuman-in-the-Loop(HITL)を組み合わせた決定論的制御フローが不可欠である¹。

特に知財領域においては、出願前の発明情報という最高度の営業秘密を取り扱うため、外部の商用クラウドAPIへデータを送信できないという厳格な制約が存在する⁹。この「プライバシー保護(ローカル実行の要請)」と「高度な推論能力(巨大モデルの要請)」のトレードオフを打破する手段こそがハーネス設計である¹¹。事実、発明提案書から明細書とクレームを一貫生成するベンチマーク(Dis2Pat)において、役割分解されたマルチエージェント・ハーネス「Patent-MAF」は、外部APIを一切呼び出さないローカル実行のオープンソースモデル群のみで構成されながら、総合品質スコアで85.4、BERTScoreで82.3、BERT-for-Patent意味的類似度で96.1を記録し、単一の巨大クロードモデルであるGPT-5(総合品質85.1、BERTScore 81.8、BERT-for-Patent類似度96.4)やオープンソースのベースラインであるQwen3-VL-72B(総合品質81.8)およびQwen3-VL-32B(総合品質81.3)を凌駕または同等以上の性能を実証している¹¹。すなわち、知財AIエージェントの導入においては、モデルの選定以上にハーネス・エンジニアリングが投資対効果と法的安全性の双方を支配する¹。

評価対象システム / ハーネス介入指標	構成・入力モダリティ	総合品質スコア (0-100)	BERTScore (%)	BERT-for-Patent 類似度 (%)	ハーネス設計に伴うシステム変動効果
Patent-MAF (マルチエージェント)	ローカル実行OSS / 役割分解ハーネス ¹¹	85.4 ¹¹	82.3 ¹¹	96.1 ¹¹	外部API通信ゼロで商用フロントエンドモデルを上回る総合品質を達成 ¹¹
GPT-5 (クロード・フロンティア)	テキスト+図面(マルチモーダル入力) ¹¹	85.1 ¹¹	81.8 ¹¹	96.4 ¹¹	テキストのみ入力時(類似度95.7%)から図面追加により+0.7pt改善 ¹¹
Qwen3-VL	テキスト+	81.8 ¹¹	—	—	テキストの

-72B (オープンソース単体)	図面(マルチモーダル入力) ¹¹				み入力時(81.1)から図面グラウンディングにより+0.7pt改善 ¹¹
Qwen3-VL-32B (オープンソース単体)	テキスト+図面(マルチモーダル入力) ¹¹	81.3 ¹¹	—	—	テキストのみ入力時(81.0)から図面グラウンディングにより+0.3pt改善 ¹¹
Vibe Patenting (QA Judge 反復改訂)	構造化QA フィードバック・ループ統合 ¹³	—	—	—	改訂ラウンドを通じ平均**+1.85ポイント**の品質向上(軽量モデルが高推論機に迫る) ¹³
SWE-bench ハーネス 構造変更	同一モデル重み下での再試行・圧縮変更 ²	—	—	—	タスク解決率が20パーセントポイント以上変動 ²
SafeHarness セキュリティ統合	4フェーズ・ライフサイクル防御統合 ¹⁵	—	—	—	危険行動率(UBR)を約38%低減、攻撃成功率(ASR)を約42%低減 ¹⁵

2. 知財法制・ガイドラインが規定するハーネス境界条件

知財業務向けハーネスは、一般的なビジネスワークフローと異なり、日本弁理士会の「弁理士業務AI活用ガイドライン」(2025年4月策定)や米国特許商標庁(USPTO)の「AIツール利用に関するガイダンス」(2024年4月公表・89 FR 25609、および2025年11月発明者適格性ガイダンス)などの法的・倫理的規範を、システムアーキテクチャの「ハード・ガードレール」としてコード化しなければならない⁴。単にプロンプトの注意書きで安全性を指示するだけでは不十分であり、権限管理、通信経

路、監査ログ、承認ゲートをハーネスとランタイムの双方で構造的に強制する必要がある⁶。

新規性喪失リスクと機密保持（データ主権の確保）

出願前の発明内容や明細書ドラフトを、秘密保持規約やオプトアウト（再学習禁止）保証のない第三者のAIプラットフォームへ入力する行為は、特許法上の「公然知られた発明」とみなされ、新規性喪失（特許無効事由）を引き起こす致命的リスクを孕む⁹。また、クライアントから委託を受けた案件では守秘義務違反に直結し、個人情報が含まれる場合に外国のAI事業者へデータが送信されれば個人情報保護法上の越境移転規制（本人同意の取得等）にも抵触し得る¹⁷。さらに米国実務（USPTOガイダンス）においては、AIツールのサーバーが米国外に配置されている場合やデータ漏洩が生じた場合、国家安全保障上の輸出管理規制（Export Control）や外国出願許可（Foreign Filing License）違反に問われるリスクが明示されている¹⁸。

したがって、ハーネスのワークスペース層およびランタイム層では、案件の機密区分や技術属性（防衛・輸出管理対象技術等）に応じて、完全オンプレミスやローカル環境（Patent-MAF型構成）へのルーティング強制、データ保持ゼロ（Zero-Data Retention）かつモデル学習への転用を契約上・技術上遮断した専用エンドポイントへの限定、ならびに案件単位でのAI利用ブロックやリージョン固定ポリシーの適用を系統的に強制するゲートウェイを実装しなければならない⁴。

善管注意義務・署名責任と庁ITシステムへの直接アクセス禁止

日本弁理士会は、生成AIの出力は正確性が保証されたものではなく「判断の材料」に過ぎないとし、弁理士がAIの生成結果を十分に検討・確認せずにそのまま出願等の代理に使用した場合、民法第644条の善管注意義務違反に問われるおそれがあると警告している³。同様にUSPTOも、提出書類がいかんにして生成されたかにかかわらず、37 CFR 11.18(b)に基づく署名者の認証責任（事実的根拠および法的妥当性の合理的調査義務）が全面的に適用されることを明確化している⁴。たとえば、AIエージェントがハルシネーションによって実在しないMPEP（審査便覧）の条項や判例を引用した場合、あるいは先行技術文献（IDS）の中身を人間が確認せずに自動提出した場合、署名した実務家が制裁対象となる⁴。また、日本弁理士会のガイドラインは、AI生成物が他者の知的財産権、とりわけ著作権（複製権や利用許諾）を侵害していないかの確認を求めているほか、米国実務においても人間の創作的寄与なしにAI単独で生成された著作物はパブリックドメインに帰属し著作権保護を受けられないことが確定しているため、この観点からも生成過程における人間の関与の検証が不可欠となる¹⁷。

さらに極めて重要なインフラ制約として、USPTOのアカウント（USPTO.gov）は自然人に限定されており、AIシステムがアカウントを取得することや、書類に電子署名を行うこと、あるいはAIエージェントが自律的にUSPTOのITシステムへアクセスしてスクレイピングや自動提出を行うことは厳格に禁止されている¹⁸。承認された制限を超えるネットワークトラフィックはUSPTOによって常時監視されており、違反時にはアカウント剥奪のみならず民事・刑事・行政上の処罰が科される¹⁸。加えて、Thaler v. Vidal判決および2025年11月のUSPTOガイダンスが確認した通り、AIは特許法上の発明者（Inventor）になり得ず、発明の成立（Conception）には人間による同時期の認識と評価（Contemporaneous recognition and appreciation）が不可欠であるため、AI支援発明においては人間による創作的寄与のプロセスを証跡として記録・保持する機能がハーネスに求められる¹⁹。

法規制・ガイドライン上の要請	根拠規範・ガイドライン	知財実務における具体的リスク	ハーネスおよびランタイムで実装すべき
----------------	-------------	----------------	--------------------

			技術的制御
新規性喪失の防止と 秘密保持	日本弁理士会ガイド ライン / 各国特許法 新規性規定 ⁹	第三者AIへの再学 習による発明の公知 化・特許無効化、守 秘義務契約(NDA) 違反 ⁹	ゼロデータ保持(No-Training) APIの 強制、ローカルLLM(Patent-MAF等)への フォールバック、PII/ 機密フィルタ ⁴
輸出管理・越境デー タ移転規制	USPTO AIガイダンス Section III(D) / 個人 情報保護法 ¹⁷	国外サーバー経由 による輸出管理規制 違反、外国出願許可 違反、個人データ越 境移転違反 ¹⁷	ワークスペース単位 での実行リージョン 固定、特定技術案件 における外部クラウド 通信の物理的遮断 ²⁴
善管注意義務と署名 者の検証責任	民法第644条(弁理 士会) / USPTO 37 CFR 11.18(b) ³	架空の判例・公報引 用、未確認IDSの提 出による懲戒処分や 権利行使不能 ⁴	全ての引用番号・条 文を序DB原文と突 合する決定論的検 証フック、起案者と承 認者を分離したHITL 承認ゲート ⁹
庁ITシステムの保護 と自然人主義	USPTO AIガイダンス Section III(C) ¹⁸	AIによる自動出願や 過剰APIアクセスによ るUSPTOアカウント 剥奪・法的制裁 ¹⁸	エージェントからの 直接提出(Filing) ツールの排除、レー トリミット制御と人間 による最終提出操作 の強制 ¹
発明者適格性と著作 権侵害防止	USPTO 2025年11月 指針(Thaler判決) / 弁理士会指針 ¹⁷	AI単独生成物の権 利不成立(パブリック ドメイン化)、他者著 作物の無断複製 ¹⁹	人間のプロンプト指 示・修正履歴(創作 的寄与の証跡)のイ ミュータブルな監査 ログ保存、類似度検 査 ¹⁹

3. 業務プロセス別・ハーネス設計の勘所と失敗モードの抑止

知財業務は「調査・分析」「明細書・クレーム起案」「中間処理(OA応答)」で要求される認知プロセスと失敗モードが全く異なるため、単一の汎用エージェント・ループで処理してはならない²。タスク特性に応じた専用ハーネスの設計が求められる⁸。

3.1 先行技術調査・無効資料調査: コンテキスト劣化の防止とAPIクォータ制御

先行技術調査において、LLMは「同義語の展開」「長文の要約」「構成要素への分解」「指定基準での一次スクリーニング(ランキング)」といった言語処理には高い適性を示す一方、「特定の公報が実在するか」「権利が存続しているか(法的状態)」「調査に漏れがないか(網羅性)」という事実確認(ファクトチェック)を極めて苦手とする⁹。また、検索における再現率(Recall)100%を確保するために10万件の母集団をAIでランキングさせるアプローチは有効であるものの、ハーネスの設計を誤ると深刻な破綻が生じる²⁷。

第一の注意点は、コンテキストの劣化(Context Degradation)と公報間の混同防止である⁹。一度のプロンプトに多数の公報テキストを詰め込むと、モデルが後半の公報の扱いを雑にしたり、異なる公報の技術要素を混ぜ合わせて架空の先行技術を合成したりする現象が頻発する⁹。ハーネスのコンテキスト管理層(*C*)は、検索母集団を数件単位のマイクロバッチに分割して並列または逐次処理させ、厳格なJSONスキーマ検証(Output Parsing & Schema Validation)を通じて「公報番号・対比結果・判定理由」を定型フォーマットで出力させるMap-Reduce型のパイプラインを組む必要がある⁶。さらに、AIが「関連なし」と判定した文献群に対しても、人間が調査目的と予算に応じた一定割合を抜き取り検査(サンプリング監査)し、食い違いが1件でも発見された場合はAIの判定理由ログを遡及検証して人間が読む範囲を広げるUI/UXをハーネスに組み込まなければならない⁹。また、調査の網羅性を事後的に証明できる唯一の根拠となるため、使用した検索式、実行日付、ヒット件数を決定論的にステート管理層(*S*)へ記録することが必須である⁶。

第二の注意点は、特許庁APIのレートリミットとクォータ枯渇の制御である²⁹。AIが示した公報番号、公開日、優先日、および法的状態は、必ずJ-PlatPatや欧州特許庁(EPO)のOpen Patent Services(OPS)等の公式データベースAPIを介して原文・原簿データと1件ずつ決定論的に照合しなければならない⁹。しかし、各庁のAPIには厳格なインフラ制約が存在する。

特許データソース / API	クォータ・取得件数上限	上限超過時のペナルティ・挙動	ハーネス層で実装すべきスロットリング・キャッシュ戦略
J-PlatPat 特許情報取得API	API種別ごとに1日200回(民間事業者は申請により500回、合理的理由で最大2倍まで緩和可能) ³⁰	1日のアクセス上限到達時点で翌日まで当該APIの利用が完全停止 ³⁰	自律ループによる直接呼び出しの禁止、社内DBキャッシュの優先参照、日次トークンバケットによるクォータ保護 ⁶
EPO OPS API (v3.2)	クエリあたり最大2,000件。無償枠は週4GBまで(有償枠は年額2,800ユーロ) ²⁹	ファミリー照会(/family)は1ファミリーごとに1コール消費。Fair Useポリシー違反時は遮断 ²⁹	APIレスポンスヘッダからのクォータ残量リアルタイム監視、書誌データと全文取得の段階的分離 ²⁹
USPTO Patent	自然人アカウント限	認可上限を超える自	スクレイピングツール

Center	定・ネットワークトラフィックの常時監視 ¹⁸	動アクセスに対しアカウント剥奪および民事・刑事・行政罰 ¹⁸	ルの完全排除、公式バルクデータまたは認可済み商用特許DB APIへの迂回ルーティング ¹⁸
--------	-----------------------------------	---	--

もしエージェントが未制御のループ内で「検索結果が不十分であるためクエリを変えて再取得する」という動作を繰り返せば、J-PlatPatの1日200～500件の枠は数分で枯渇し、特許事務所や知財部全体の業務システムが翌日まで停止する事態を招く⁶。ハーネスのツール・ディスパッチ層(T)には、APIレスポンスヘッダからクォータ消費量を読み取る監視機構、同一公報への重複アクセスを防ぐローカルキャッシュ、およびステップ上限(Max-Step Limits)と予算キャップを必ず設定しなければならない⁶。

3.2 明細書・クレーム起案: マルチエージェント分解とマルチモーダル図面グラウンディング

実務における明細書ドラフティングの最大の難所は、発明者が記述した非法律的で断片的な発明提案書(Invention Disclosure)と図面から、法的に構造化されたクレーム群と詳細な説明を一貫性をもって立ち上げるプロセスにある¹¹。単一のLLMIに提案書を入力して一度に明細書全文を出力させるモノリシックな生成は、構成要素の抜け漏れやクレームと明細書の不整合を引き起こすほか、生成された要約や本文が入力の冗長性をそのまま引きずり、定型的・反復的な表現に偏る傾向がある¹¹。最新のベンチマーク研究(Patent-MAFおよびVibe Patenting)は、ハーネスを3つの専門的役割に分解するマルチエージェント・オーケストレーションの有効性を立証している¹¹。まず第一段階として「マネージャー・エージェント(Manager Agent)」が起動し、非法律的テキスト T と図面群 F からなる発明提案書 $D = (T, F)$ を受け取って、いきなり文章を起案するのではなく、明示的な出力スキーマに従って2種類の構造化中間データ $M = (M_C, M_S)$ へと変換する¹¹。ここで M_C は権利範囲を画定する法的に不可欠な特徴を抽出したクレーム起案用マテリアルであり、 M_S は実施形態や技術的效果を整理した明細書起案用マテリアルである³³。続く第二段階では、マネージャーが出力した共通表現 M_C, M_S に厳密にグラウンディングされた状態で複数の「専門ドラフター・エージェント群(Specialized Drafter Agents)」が独立して並列稼働し、少なくとも1つの独立クレームと複数の従属クレームを含むクレーム群 C と、背景技術・図面の簡単な説明・実施形態等の明細書セクション S をそれぞれ生成する³³。この際、視覚エンコーダ(Vision Encoder)を介した図面入力の統合が極めて重要となる¹¹。テキストのみのモデルでは部材間の空間的配置や接続関係を誤記するケースが多発するが、図面をマルチモーダル入力としてドラフターへ供給することで空間的・構造的記述の完全性が向上し、実際にQwen3-VL-72Bで総合品質スコアが81.1から81.8へ、GPT-5のBERT-for-Patent類似度で95.7%から96.4%へ改善することが確認されている¹¹。そして最終段階として「ポリッシャー・エージェント(Polisher Agent)」が各ドラフターの出力を統合し、用語の統一性、先行詞の整合性、およびクレームと明細書の法的整合性を洗練させて最終ドラフト $P = (C, S)$ を完成させる¹¹。

また、複数のセッションにまたがる長大な明細書作成プロジェクトにおいては、コンテキストウィンドウが10万～100万トークンに拡大した現代のモデルであっても「セッション間の健忘 (Amnesia)」やコンテキスト劣化が生じる²⁰。これを防ぐため、初回セッションで一度だけ起動して発明の検証可能な構成要素リスト (feature_list.json 等) と進捗ログファイルを環境としてセットアップする「イニシャライザ・エージェント (Initializer Agent)」と、後続セッションで毎回クリーンなコンテキストから立ち上がり、状態ファイル群を即座に読み込んで1つの実施形態やクレーム群のみを段階的に実装・テスト検証してコミットする「実行エージェント」に分割する2エージェント・ハーネス設計が極めて有効である³⁵。さらに、日本特許庁 (JPO) の審査基準およびAI関連技術事例集に照らし、AIによってある機能を持つと推定された物の発明 (マテリアルズ・インフォマティクスや創薬等) の明細書を起案する場合、ハーネスは特別な検証ルールを適用しなければならない³⁶。JPOの実務では、実際に物を製造して機能を評価した実施例が明細書に記載されていない場合、複数種類のデータ間に相関関係等が存在することが明細書の記載または技術常識から推認できること、AIの示す予測値の予測精度が明細書等で検証されていること、あるいはAIによる予測結果が実際に製造した物の評価に代わり得るとの出願時の技術常識があることのいずれかが満たされなければ、実施可能要件およびサポート要件違反として拒絶される³⁶。したがって、起案ハーネスは発明提案書に実施例データやAI予測精度の検証データが不足している場合、ハルシネーションで架空の実験データを捏造することを厳禁とし、発明者へ追加の検証データを要求するエスカレーション・分岐を発動させなければならない⁶。

3.3 拒絶理由通知 (OA) 応答: 5大失敗モードの阻止

オフィスアクション (拒絶理由通知) への応答書・補正書作成は、1語の相違が権利範囲を左右し、正確な引用とピンポイントの根拠提示が求められるため、言語モデルにとって最も過酷なタスクの一つである²⁴。実務上、AIによるOA応答ドラフティングでは5つの典型的な失敗モードが繰り返し発生する²⁴。第一の失敗モードは「引用文献に存在しない架空の抜粋 (Fake Quotations)」であり、審査官が引用した先行技術文献 (引例) の該当箇所を要約・反論する際、引例には存在しない文言を鉤括弧付きで捏造する事象である²⁴。第二の失敗モードは「クレームに記載されていない特徴に基づく反論 (Arguing Unclaimed Features)」である²⁴。LLMは本願発明と引例が技術的にどう異なるかを説明することには長けているが、その相違点が現在の請求項 (クレーム) の文言として実際に記載 (Recite) されているか無頓着であり、クレームにない特徴を根拠とする進歩性・非自明性の主張は審査官によって即座に退けられる²⁴。第三の失敗モードは「当初明細書にサポートのない補正 (新規事項の追加 / New Matter)」であり、拒絶を回避するためにAIが提案した補正クレームが文脈上いかに自然に見えても、出願当初の明細書および図面に書面による説明 (Written Description Support) がなければ、米国特許法第132条(a)や日本特許法第17条の2第3項に違反する新規事項の追加として新たな拒絶理由を招く²⁴。第四の失敗モードは「拒絶理由自体の読み違い (Misreading the Rejection)」であり、審査官が認定した拒絶の法的根拠や請求項ごとの適用関係を誤読して応答すべき論点を落とす事象である²⁴。そして第五の失敗モードは「様式違反および機密漏洩」であり、37 CFR 1.121等が定める補正の下線 (追加)・取消線 (削除) やステータス表示のフォーマット崩れにより実務家が手作業で書き直す手戻りが発生するほか、未公開クレームや輸出管理対象技術を汎用チャットボットへ入力することによる機密流出が生じる⁴。

これらを抑止するため、OA応答用ハーネスは、モデルが反論を生成する前に「審査官の拒絶理由の構造化マッピング」を状態として固定し、モデルが提案する全ての補正文言に対して「当初明細書の段落番号・原文との完全一致または直接的サポート」を強制的にリンクさせる検証ステップを挟まなければならない⁶。また、主張される全ての相違点が、補正後のクレームの構成要件 (要素) のどの

文言に対応するかを1対1でマッピングするテーブルを内部状態として生成させ、マッピングが存在しない反論は出力前にハーネスが棄却・再生成(Retry)させる制御が必要である⁶。

4. 検証ゲート(Verification Gates)の二重化: 決定論的リンターとLLM-as-a-Judgeの統合

ハーネスの検証層(V)を構築する際、近年主流となっている「LLM-as-a-Judge(評価専用LLMエージェントによる自己修正ループ)」のみに依存することは、知財実務において極めて危険である¹³。

エンドツーエンドの特許起案評価基盤である「Vibe Patenting」(2026年)の実験によれば、独立したQA Judgeエージェントを反復的な改訂ループ(Iterative Revision Loop)に組み込み、5回の独立評価の平均スコアと構造化されたフィードバックを与えることで、通常の自己修正では飽和してしまうドラフト品質を継続的に向上させ(平均+1.85ポイントの改善)、推論コストの低い軽量モデルを、はるかに高価な高推論モデル(Extra-High Reasoning)に匹敵するスコアへ引き上げることが可能である¹³。しかし同時に、同研究が人間の熟練弁理士による評価とLLM Judgeの評価を同一の5つの品質次元で突き合わせた結果、両者の一致度は評価次元(メトリクス)に強く依存し、体系的なスコアバイアスや相関の乖離が存在することが判明している¹³。すなわち、自動評価エージェント(LLM Judge)に対する最適化スコアが向上したからといって、それが直ちに弁理士視点での法的有効性や実務品質の向上を意味するわけではない¹³。

LLMは確率的トークン生成器であるため、構文的な網羅性や文体の流暢さを評価することには適しているが、厳密な形式論理や数値・記号の一致検証には向かない²。したがって、知財ハーネスの検証ゲートは、「決定論的パテント・リンター(Deterministic Patent Linters)」と「確率論的QA Judge(LLM-as-a-Judge)」の二層構造として設計しなければならない¹⁴。

第一層: 決定論的パテント・リンター(構造テストとフック)

ソフトウェア開発用エージェントにおいてカスタムリンターや構造テストが厳格な依存関係レイヤー(

Types → Config → Repo → Service → Runtime → UI)を強制し、すべてのアクションがエンジニアの制御するフック(Hook)を通過するのと同様に、知財ハーネスではルールベース検証エンジン(plint、ClaimMaster、Patent Bots等に相当する決定論的スクリプト)をツール呼び出し後・出力前のフックとして強制実行する必要がある³⁹。具体的には、まずクレーム内で「前記～(the / said)」として参照される名詞句が、それより前の箇所または引用元の親クレームで初出要素として正しく導入されているかを構文木ベースで完全検証する「先行詞(Antecedent Basis)の自動検査」を実行する⁴⁰。同時に、多重従属クレームの引用ルール違反やクレーム番号の欠番、キャンセルされたクレームへの参照を検知する「クレーム従属関係と番号体系の整合性チェック」を行う⁴⁰。さらに、クレーム内のすべての技術用語やフレーズが発明の詳細な説明内に少なくとも1回以上出現しているか(Word / Phrase Support)、および図面中の参照符号(Part Numbers)と明細書本文中の符号・名称の対応に不一致や未定義がないかを決定論的にクロスチェックする⁴⁰。OA応答や調査報告においては、AIが抜粋した引用文が引例のOCRテキストやXML原文の該当段落に一字一句違わず実在するかを部分文字列検索で検証し、最終的な庁提出フォーマット(USPTOの37 CFR 1.121に基づく追加箇所の下線・削除箇所の取消線・クレーム一覧と意見書セクションの分離等)への変換も、LLMではなく決定論的テンプレートエンジンで処理させる⁴。

第二層: 確率論的QA Judgeと失敗知識の動的蓄積

第一層の決定論的リンターをすべて通過(Pass)したドラフトに対してのみ、第二層のQA Judgeエージェントを起動し、「発明の本質的特徴の網羅性」「明瞭性(Clarity)」「適法性(Legality)」「クレームと明細書の一貫性」や、商業製品から省略可能な構成要素の特定・置換シミュレーションによる「設計回避(Design-around)に対するクレームの堅牢性」といった高次の意味論的評価を実施する¹¹。さらに、コーディングエージェントにおける AGENTS.md のベストプラクティスを応用し、実務家(弁理士)がレビュー時に修正・却下したパターンや特定のクライアント固有の記載ルールを、静的なドキュメントではなく「生きた制約システム(Living Constraint System)」としてリポジトリまたはワークスペース設定ファイルへ継続的に追記し、次回セッション起動時にハーネスが自動的にコンテキストへ注入する動的フィードバックループを構築すべきである²⁰。

対象業務フェーズ	典型的なAI失敗モード	第一層: 決定論的リンター(コードによるハード検証)	第二層: 確率論的QA Judge(LLMによる意味検証)	最終ゲート: Human-in-the-Loop(人間による承認)
先行技術・無効資料調査	実在しない公報番号の生成、法的状態の誤認、後半公報の評価漏れ ⁹	J-PlatPat / EPO OPS APIとの公報番号・書誌・存続状態の自動照合、JSONスキーマ検証 ⁶	検索クエリと発明構成要件の意味的対応度、スクリーニング除外理由の論理的妥当性評価 ⁹	「関連なし」判定公報の抜き取り監査、検索式・日付・ヒット件数の確定署名 ⁹
明細書・クレーム起案	先行詞欠如、図面符号の不一致、クレーム用語の明細書不記載、AI予測発明のサポート不足 ¹¹	先行詞チェッカー(plint等)、用語出現チェック、図面番号・符号の一致検査 ⁴⁰	発明提案書(D)からの構成要素の抜け漏れ、VLMによる図面との空間的整合性、設計回避耐性 ¹¹	弁理士による権利範囲の広狭判断、AI関連発明の相関関係・実験データの妥当性最終判断 ⁹
OA応答・補正書作成	引例の架空引用、未クレーム特徴による反論、新規事項追加、37 CFR 1.121等の様式不備 ⁴	引例原文との完全一致文字列検索、補正差分の自動生成(下線・取消線)、当初明細書内の語句検索 ⁴	全ての拒絶理由に対する反論の網羅性、主張する相違点とクレーム文言の論理的結合度 ¹⁴	起案者とは別の承認弁理士によるレビュー、37 CFR 11.18(b) / 善管注意義務に基づく署名と証跡保存 ³

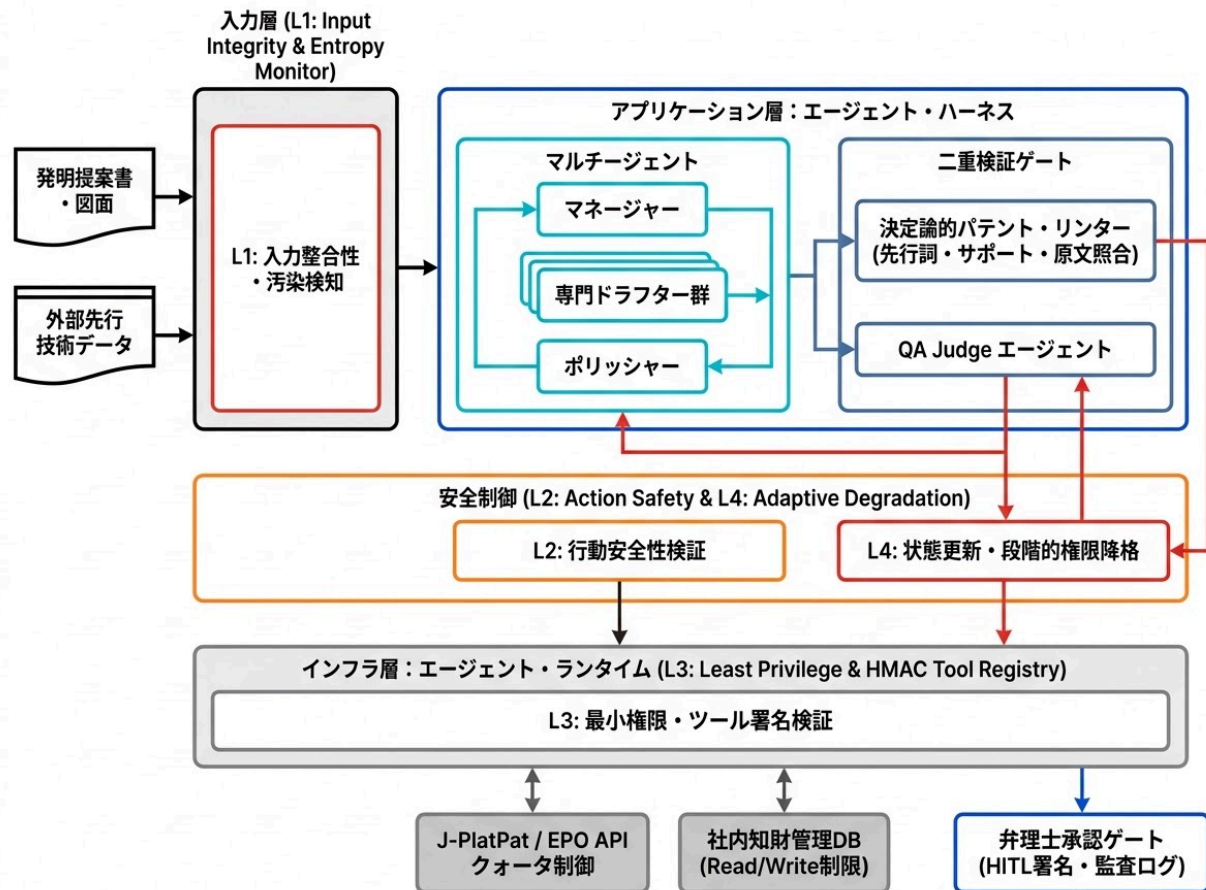
5. ライフサイクル統合型セキュリティと耐障害性(

SafeHarnessの知財展開)

知財AIエージェントは、社外の不特定多数のデータ(競合他社の公開特許公報、非特許文献、技術ウェブページ)を検索・取得して読み込みながら、同時に社内の最高機密データベース(未公開の発明提案書、知財戦略、知財管理システム)や外部APIへアクセスするツール権限を持つ²⁰。この構造は、外部文献に埋め込まれた悪意ある文字列や汚染されたメタデータによってエージェントの推論が乗っ取られる「間接プロンプトインジェクション(Indirect Prompt Injection)」や「RAG汚染」、およびそれに伴う「ツール悪用(Tool Misuse / 機密データの外部送信やDB破壊)」の格好の標的となる⁷。

従来、多くの企業システムでは、NeMo GuardrailsやLlama Guardのような入出力フィルタをモデルの「外側」に後付け(Bolt-on)する対策が採られてきた⁷。しかし、近年のエージェント安全性研究(Lin et al., 2026『SafeHarness』)は、こうした外付け型ガードレールには3つの構造的欠陥があり、自律型エージェントの保護には不十分であることを明らかにしている⁷。第一の欠陥は「コンテキスト盲目性(Context Blindness)」であり、外付けガードレールは会話インターフェースの入出力のみを検査してハーネス内部の状態(中間推論ステップやツール観察結果)を認識できないため、先行技術検索ツールが取得した外部テキストに汚染された観察結果(Poisoned Tool Observation)が含まれて推論チェーンに入り込み後続のツール呼び出しを誘導しても、呼び出し単体しか見ない外側のガードレールは攻撃の因果関係(Adversarial Provenance)を検知できない⁷。第二の欠陥は「レイヤー間孤立(Inter-layer Isolation)」であり、入力フィルタ、行動検証器、メモリ管理が互いに孤立して動作しているため、入力層で不審なシグナルを検知しても下流のツール実行検証器の警戒レベルを引き上げたり、ブロックされたツール呼び出しに基づいてメモリサブシステムに当該コンテキストの不信フラグを伝達したりする連携機構が存在せず、入力・ツール出力・会話履歴を同時に狙う複合攻撃(Composite Attacks)に対して無力となる⁷。第三の欠陥は「レジリエンスの欠如(Lack of Resilience)」であり、既存の防御は通過(Pass)か遮断(Block)かの二値判定しか持たないため、一度攻撃が外周防御を突破するとエージェントのツール権限を段階的に縮小(Graceful Degradation)させる仕組みがなく、汚染されたコンテキストのまま後続ステップが実行されて被害が雪だるま式に拡大する⁷。

知財AIエージェントのハーネス設計においては、セキュリティ機構を外付けするのではなく、ハーネスの実行ライフサイクル(「入力処理 → 意思決定推論 → アクション実行 → 状態更新」の4フェーズ)の第一級オブジェクトとして直接統合する「SafeHarness」アーキテクチャの採用が強く推奨される⁷。実証実験において、SafeHarnessはタスクの実用性を維持したまま、保護なしのベースラインと比較して危険行動率(UBR: Unsafe Behavior Rate)を約38%、攻撃成功率(ASR: Attack Success Rate)を約42%低減させている¹⁵。



知財システムへの具体的なマッピングは、L1からL4までの4つの防御層と層間フィードバックによって構成される²⁸。

L1(INFORM: 入力整合性とエントロピー監視)

ユーザー入力や外部から取得した公報・ウェブテキストがエージェントのコンテキスト(C_k)に入る前に、パターンフィルタリングと意味的サニタイズを実行する¹⁵。もし外部公報テキスト内に間接プロンプトインジェクションの兆候が検知された場合、エントロピー監視器(Entropy Monitor)が違反を記録し、累積エントロピードリフトが閾値を超えた時点で即座にL2層の最小検証ティアを引き上げる(`l2.escalate_verification()` および `_last_untrusted` フラグによる `l2.set_min_tier(2)` の発火)²⁸。これにより、単なる入力遮断にとどまらず、後続の推論ステップ全体が高い警戒レベルで処理される層間フィードバック(L1 → L2)が成立する¹⁵。

L2(VERIFY: 行動安全性と因果診断)

たとえ入力がクリーンであっても、エージェント自身がハルシネーションや推論ミスによって危険な意思決定(未承認での社内DB書き換えや機密情報の外部検索クエリへの混入など)を行う確率は22～56%に達することが報告されている²⁸。L2層はエージェントが生成したツール呼び出し要求

(t_i, a_k) を実行前に独立検証する¹⁵。たとえば、先行技術検索用の外部API(J-PlatPatやGoogle Patents等)へ投げる検索クエリ引数 a_k の中に、出願前の未公開クレーム全文やクライアント固有

の秘密パラメータがそのまま含まれていないかをルールおよび因果診断 (Tier 3 Causal Diagnosis) で検査し、情報漏洩を未然にブロックする⁹。

L3(CONSTRAIN: 最小権限の原則とHMACツール署名)

ツール定義の改ざん (Tool Description Tampering) や権限逸脱を防ぐため、ハーネスのツール・レジストリ (T) に登録されるすべてのツールにHMAC署名を付与して実行時に検証するとともに、ツールを5段階のリスク体系 (destructive $\rightarrow$ network $\rightarrow$ execute $\rightarrow$ write $\rightarrow$ read_only) に厳密に分類し、ケイパビリティ・トークンによる最小権限 (Least Privilege) を強制する²⁸。知財システムにおいては、社内知財管理システム (IPMS) のレコード削除や庁への直接提出 (destructive) はエージェントから完全に除外し、外部ネットワーク通信 (network) やドラフト保存 (write) にも厳格なトークン制約を課す¹⁸。

L4(STATE UPDATE: 状態保護と適応的権限降格)

SafeHarnessの最も重要な特徴が、L4層による段階的な権限降格 (Graceful Privilege Degradation) とメモリ保護である¹⁵。L4の降格レベル (0 = 通常稼働 $\sim$ 4 = 完全読み取り専用) はL3の5段階リスク体系と1対1で対応している²⁸。実行中に不審な挙動や検証エラーが蓄積した場合、ハーネスはシステムをいきなりクラッシュさせるのではなく、降格レベルを引き上げて write や network 権限を動的に剥奪し、read_only (既存の取得済み公報のローカル分析のみ許可) へと段階的に縮退運転させる⁷。その後、連続して安全な操作が規定のウィンドウ (recovery_window) に達した段階で、l4.maybe_recover() および l2.maybe_deescalate() によって段階的に権限と検証ティアを通常状態へ復帰させることで、セキュリティと業務継続性 (ユーティリティ) の最適バランスを実現する²⁸。

6. 結論と段階的導入ロードマップ

知財業務へのAIエージェント導入において、ハーネス設計は単なるラッパー開発ではなく、「確率的なAI推論」と「決定論的な知財法制・実務規範」を架橋する基幹インフラの構築に他ならない²。本分析から導かれる知財ハーネス設計の鉄則と段階的導入アプローチは以下の通りである。

第一に、いきなり全社・全業務へ自律型エージェントを展開しないことである²¹。初期フェーズでは、特定の技術テーマや専任チームに絞り込み、かつ定型的で繰り返しの多い単一業務 (例: 公開公報の数件単位での構成要素対比マトリクス生成や、発明提案書からの図面符号リスト・先行詞チェック等) のみを固定ワークフロー (DAG) としてハーネス化するスモールスタートが不可欠である²。トライアル期間中であっても、法務部門・情報システム部門・知財部門が連携し、外部APIのデータ保持期間、暗号化基準、SOC2やISMS等の認証有無、およびアクセス権限設定が社内規程や守秘義務契約に抵触しないかを精査しなければならない¹⁷。

第二に、ローカル実行可能なマルチエージェント構成と決定論的リンターの組み合わせを標準アーキテクチャとすることである¹¹。出願前の発明を取り扱う起案業務では、Patent-MAFが実証したように、提案書の構造化 (マネージャー)、図面を伴う並列起案 (ドラフター)、整合性洗練 (ポリッシャー) へと役割を分解することで、外部クラウドへ機密を送信せずとも商用フロンティアモデルと同等以上の起案品質を確保できる¹¹。そして、生成されたクレームと明細書に対しては、LLM Judgeによる意味的レビューの前に、必ず先行詞・用語サポート・符号一致・原文引用一致を検査する決定論的リンターをハード・ゲートとして通過させなければならない¹⁴。

第三に、起案と承認の分離、および完全な監査証跡 (Observability) の実装である²⁰。どれほどハーネスが高度化しようとも、日本弁理士会ガイドラインおよびUSPTO 37 CFR 11.18(b)の下で最終的な

法的責任を負うのは署名する自然人(弁理士・弁護士)である⁴。ハーネスは、AIがドラフトを生成・修正したプロセスと、それを人間がレビュー・承認するプロセスを明確に分離し、「どのバージョンの出力に対し、どの検索式と原文照合結果に基づき、誰が、いつ、いかなる理由で承認署名を行ったか」を再構築可能な監査ログとしてイミュータブルに記録するガバナンス基盤として機能しなければならない⁹。これら決定論的制御・二重検証ゲート・ライフサイクル統合型セキュリティ(SafeHarness)を備えたハーネスを構築して初めて、知財組織は新規性喪失や善管注意義務違反のリスクを統制しつつ、AIエージェントによる生産性革新を享受することが可能となる¹。

引用文献

1. What is an AI Agent Harness? | Databricks Blog,
<https://www.databricks.com/blog/ai-harness>
2. The Architectural Evolution of LLM Agent Harnesses - Ugur Candan,
<https://ugurcandan.com/the-architectural-evolution-of-llm-agent-harnesses/>
3. AI 等を用いた業務支援サービスの提供と 弁理士法第 75 条との関係,
<https://www.jpaa.or.jp/cms/wp-content/uploads/2025/04/AIservices-article75.pdf>
4. PatentSolve vs Solve Intelligence for Office Action Response (2026),
<https://patentsolve.com/blog/patentsolve-vs-solve-intelligence-office-action-response>
5. The Infrastructure Layer That Makes AI Agents Actually Work,
<https://dev.to/vishalmysore/harness-engineering-the-infrastructure-layer-that-makes-ai-agents-actually-work-4nl1>
6. Agent Harness vs Agent Runtime: What's the difference? - Credal,
<https://credal.ai/blog/agent-harness-vs-agent-runtime>
7. SafeHarness: Lifecycle-Integrated Security Architecture for LLM,
<https://arxiv.org/html/2604.13630v1?ref=swarmsignal.net>
8. Agent Frameworks vs Runtimes vs Harnesses: The AI Agent Stack,
<https://www.analyticsvidhya.com/blog/2025/12/agent-frameworks-vs-runtimes-vs-harnesses/>
9. 特許調査のAI活用 | 出願前の発明を守る使い方と手順,
<https://coplus-one.jp/tokkyo-chosa-ai/>
10. 弁理士が生成AIを知財調査・明細書作成補助に活用する際の留意点と,
<https://www.enetmercury.com/patent-attorney-ai-chizai-chosa-guide/>
11. Benchmarking Patent Drafting from Inventor-Style Disclosures - arXiv,
<https://arxiv.org/html/2608.21249v1>
12. AIハーネス・ガードレール | 企業導入の設計原則2026 - Zenn,
<https://zenn.dev/creationline/articles/ai-harness-guardrails-enterprise-2026>
13. Evaluating LLM Judges for Professional Patent-Drafting Agents - arXiv,
<https://arxiv.org/pdf/2609.13422>
14. Evaluating LLM Judges for Professional Patent-Drafting Agents - arXiv,
<https://arxiv.org/html/2609.13422v2>
15. SafeHarness: Lifecycle-Integrated Security Architecture for LLM,
<https://arxiv.org/html/2604.13630v1>
16. SafeHarness: Lifecycle-Integrated Security Architecture for LLM,
<https://arxiv.org/abs/2604.13630>

17. 弁理士業務 AI 利活用ガイドライン,
<https://www.jpaa.or.jp/cms/wp-content/uploads/2025/04/AIservices-guideline.pdf>
18. New Guidance on Use of Artificial Intelligence-Based Tools in,
<https://www.manatt.com/insights/newsletters/intellectual-property-law/new-guidance-on-use-of-artificial-intelligence-bas>
19. YOUR AI IS ON DISPLAY - U.S. District Court - District of Oregon,
<https://ord.uscourts.gov/2026%20District%20Conference/Effective%20AI%20Panel%20-%20Your%20AI%20is%20on%20Display.pdf>
20. AIエージェントハーネスとは何かをわかりやすく解説する - Databricks,
<https://www.databricks.com/jp/blog/ai-harness>
21. AIハーネスとは？ガードレールとの違い・プロンプトとの関係を整理,
<https://molonest.com/blog/ai-harness-engineering-explained/>
22. 生成AIの企業利用でハーネス/ガードレールはどこに実装するのが,
<https://note.com/skomeda/n/ne7988a71071f>
23. 弁理士業務AI利活用ガイドライン(2025年4月公表) の評価と反響,
<https://yorozuipsc.com/uploads/1/3/2/5/132566344/2ad1da223bf80d38ed2c.pdf>
24. Using AI on Office Action Responses: What the USPTO Expects,
<https://designyourinvention.com/blog/ai-office-action-response-uspto-guidance/>
25. International Patent Office Guidance on AI Inventions,
<https://www.solveintelligence.com/blog/post/international-patent-office-guidance-on-artificial-intelligence-inventions>
26. 生成AI時代の特許関連ツールの選び方と活用のポイント ～2026知財,
<https://www.tmi.gr.jp/eyes/blog/2026/18888.html>
27. 生成AIは特許サーチャーを超えたのか？ーサマリアのベンチマーク,
<https://note.com/anozaki/n/na6a1b95b7ef6>
28. SafeHarness: Security-Aware Harness Architecture for Safe AI Agent,
<https://github.com/liu-yang-maker/SafeHarness>
29. EPO Patent Search - European Patent Office OPS API - Apify,
<https://apify.com/ryanclinton/epo-patent-search>
30. 特許情報取得APIの試行提供について,
<https://ip-data.jpo.go.jp/files/%E7%89%B9%E8%A8%B1%E6%83%85%E5%A0%B1%E5%8F%96%E5%BE%97API%E3%81%AE%E8%A9%A6%E8%A1%8C%E6%8F%90%E4%BE%9B%E3%81%AB%E3%81%A4%E3%81%84%E3%81%A6%EF%BC%88202304%E8%AA%AC%E6%98%8E%E4%BC%9A%E8%B3%87%E6%96%99%EF%BC%89.pdf>
31. Open Patent Services (OPS) | epo.org,
<https://www.epo.org/en/searching-for-patents/data/web-services/ops>
32. epo-ops - patent-dev - GitHub,
<https://github.com/patent-dev/epo-ops?ref=patent.dev>
33. Benchmarking Patent Drafting from Inventor-Style Disclosures - arXiv,
<https://arxiv.org/pdf/2608.21249>
34. PatentWriter: A Benchmarking Study for Patent Drafting with LLMs,
<https://arxiv.org/html/2507.22387v1>
35. A Paradigm Shift from Prompt Engineering to Agent Harnesses - note,
https://note.com/wayne_chang/n/n9e894a5d0540?hl=en

36. AI 技術の進展をふまえた 特許審査に関する取組の最新状況,
<https://jpaa-patent.info/patent/viewPdf/4478>
37. AI 関連技術に関する事例について,
https://www.jpo.go.jp/system/laws/rule/guideline/patent/document/ai_jirei/jirei.pdf
38. AI関連技術に関する事例の追加について - 特許庁,
https://www.jpo.go.jp/resources/shingikai/sangyo-kouzou/shousai/kijun_wg/document/17-shiryou/003.pdf
39. How to Harness Coding Agents with the Right Infrastructure | Blog,
<https://alexlavaee.me/blog/harness-engineering-why-coding-agents-need-infrastructure/>
40. ClaimMaster: Automated Patent Proofreading, Patent Drafting,
<https://www.patentclaimmaster.com/>
41. btrettel/plint - patent claim proofreader and analyzer - GitHub,
<https://github.com/btrettel/plint>
42. Agent Architecture: Building AI-Powered Development Harnesses,
<https://blakecrosley.com/guides/agent-architecture>
43. A Complete Overview of Tools for Patent Professionals - Patent Bots,
<https://www.patentbots.com/feature-overview>
44. The Complete Guide to Patent Proofreading (2026) - PatentSolve,
<https://patentsolve.com/blog/patent-proofreading-guide>
45. US20170103055A1 - Patent drafting system - Google Patents,
<https://patents.google.com/patent/US20170103055A1/en>
46. ハーネスエンジニアリングとは？ AIエージェントを支える4つの構成,
https://staff.persol-xtech.co.jp/hatalabo/it_engineer/794.html
47. AIの品質ばらつきを防ぐ「ハーネスエンジニアリング」とは？ 構成,
https://aismiley.co.jp/ai_news/harness-engineering-2026/