

AEGIS(Assured Enterprise-agent Grid for Integrated Strategy)

エンタープライズエージェント品質保証戦略グリッドの提案

HITACHI

日立製作所 研究開発グループ システムイノベーションセンタ 新原敦介 福井大輔

システム階層 (EAベース)	【前提】想定される 故障モード・リスク例	① 技術的検証・テスト (開発時・静的評価)	② 実行時制御・ガードレール (運用時・動的被害抑止)	③ 運用・体制ガバナンス (組織・ヒトによるカバー)	④ 契約・ビジネス合意 (SLA・免責事項)
L5: ビジネス・社会・フィジカル (業務プロセス、人とAIの協調、実世界への影響)	・目的未達・実世界被害 ・業務KPIの未達成、業務の意図崩壊 ・フィジカルな被害(ロボットの誤作動、事故) ・CBRN(化学兵器等)への悪用・教唆 ・法令違反、社会的信用の失墜	・業務要件テスト・シミュレーション ・ビジネスゴールに対するアライメント検証 ・レッドチーミングによる極限リスク評価 ・実証環境(デジタルツイン等)での安全性テスト	・物理的/論理的インターロック ・人命に関わる操作の物理的遮断(ハードウェア側) ・重要インフラへの自律的アクセスの論理遮断	・Human-in-the-Loopの徹底 ・最終的な業務判断への人間の介入義務付け ・AI統括責任者(CAIO)による利用可否のガバナンス ・インシデント発生時の緊急対応フロー(CSIRT)	・損害賠償と利用範囲の限定 ・特定用途(兵器・重要インフラ等)の利用禁止条項 ・万が一の損害賠償上限の設定 ・顧客側での「最終確認義務」の明記(免責)
L4: アプリ・システム (マルチエージェント、既存TA連携、UI/UX)	・システム連携・協調エラー ・エージェント間通信の汚染 ・無限ループ、自律タスクの暴走 ・既存TA(トラディショナルアプリ)のAPI不正操作 ・権限侵害、不正なツール実行	・結合・連携テスト ・マルチエージェント環境の形式検証・モデル検査 ・API呼び出しのファジングテスト ・意図しないツール使用シナリオの網羅的テスト	・権限・リソースの動的制御 ・APIの流量制限(レートリミット) ・Read/Write権限の厳密な分離(最小特権の原則) ・暴走検知時の自動強制停止(Kill Switch)	・アプリ稼働監視 ・エージェント間通信ログの監視と監査 ・定期的なシステム全体のアップデートと回帰テスト	・システムSLAの再定義 ・「自律動作」に対する可用性・応答性SLA(Service Level Agreement)の適用除外 ・連携先TAのAPI仕様変更に伴う動作不良の免責
L3: データ・知識・メモリ (RAGナレッジ、長期/短期メモリ、業務データ)	・データ汚染・機密漏洩 ・メモリ汚染による知識の破壊 ・RAGの検索精度不足、不適切な情報抽出 ・機密情報・個人情報等の意図せぬ開示 ・学習データの偏りによるバイアス助長	・データ品質・参照元の評価 ・RAGの参照元妥当性(Grounding)の自動評価 ・学習用・RAG用データの網羅性・均一性チェック ・バイアス診断ツールの適用 ・TORAK	・入出力・データ保護制御 ・プロンプトインジェクションの検知・ブロック ・個人情報/機密情報の動的マスキング(PIIスクラブ) ・ベクトルDBへのアクセス制御(ロールベース)	・データ管理プロセス ・ナレッジベースの定期的な更新・陳腐化対応 ・エージェントのメモリ(文脈)の定期的なリセット運用	・データ責任の分界 ・「顧客提供データに起因する出力の誤り」の免責(Garbage In, Garbage Out) ・入力データの学習利用可否(オプトアウト)の合意
L2: モデル・推論エンジン (LLM、ファウンデーションモデル、推論ロジック)	・推論エラー・モデル脆弱性 ・ハルシネーション、論理的推論の失敗 ・敵対的サンプルへの脆弱性 ・毒性(Toxicity)のあるコンテンツ生成 ・モデル抽出攻撃、パラメータ漏洩	・モデルベンチマーク評価 ・標準ベンチマーク(HELM等)による推論能力・安全性評価 ・モデルのロバスト性(ノイズ耐性)テスト	・出力フィルタリング ・出力の確信度(Confidence)スコアによる足切り ・有害単語・不適切表現の出力前ブロック機構	・ドリフト監視とチューニング ・コンセプトドリフト(モデル劣化)の継続的モニタリング ・LLMのバージョンアップ・ファインチューニング運用	・外部モデル依存の合意 ・第三者提供LLM(OpenAI等)の仕様変更・サービス停止への免責 ・ベースモデルに内在するバイアス・欠陥に対する免責
L1: インフラ・基盤 (クラウド環境、GPU資源、ネットワーク)	・可用性低下・サイバー脅威 ・リソース枯渇による停止 ・クラウド基盤への不正アクセス、サイバー攻撃 ・サプライチェーン攻撃(オープンソースライブラリの汚染等)	・基盤セキュリティ診断 ・インフラの脆弱性スキャン、ペネトレーションテスト ・高負荷を想定したパフォーマンステスト	・インフラ保護設定 ・WAF、DDoS保護の導入 ・コンピュータ資源のオートスケーリング、クォータ制限 ・厳格なIAM(アクセス管理)制御	・死活監視・インフラ運用 ・基盤の24時間365日死活監視・アラート対応 ・ライブラリやOSの定期的なパッチ適用運用	・可用性SLAの設定 ・クラウドインフラとしての稼働率SLA(例: 99.9%)の定義 ・サイバー攻撃被害発生時の責任分界点の明記

■グリッドの読み方

縦軸：システム階層、下位のエラーが上位の業務プロセスに波及
横軸：故障モードに対する、品質と安全性を担保するための多層のアプローチ。

理想として、①②の技術的・実行時制御で防ぎたいが、現実では防ぎきれないため、③④でカバーする構造になる。

セル：品質保証アプローチ。

それぞれ一例をあげており深掘りが必要。

■なぜAEGISが必要？

- ・エージェントシステムの品質保証を体系的に整理して、開発するシステムにおいて、それぞれのレイヤの①～④のどこで保証していくかの意思決定を行いたい。
- ・エージェントシステムの品質の議論は混迷しがち。どのセルの話をしているのか明確にしたい。

■今後の課題

- ・各階層の故障モードを詳細に洗い出し。
- ・品質保証アプローチの詳細化。