



AI とコードメンテナンス

共同知性へ

内容

1. エグゼクティブサマリー	2
2. 背景：開発チームにとって AI が不可欠になる理由	2
3. 2025 年の AI の現状	3
3.1. ANI、AGI、ASI：現在の現実	3
3.2. 単純な会話からエージェントへ：会話型、生成型、推論型、エージェント型 AI	3
4. 開発タスクにおける AI の真の実力と限界	5
4.1. 純粋なパフォーマンス：IQ テストとベンチマーク	5
4.2. 適応性テスト：AI がまだ失敗している分野	6
4.3. 信頼性、幻覚、共同知能の必要性	7
5. セキュリティ、コンプライアンス、ガバナンス：AI を導入するための前提条件	9
6. 2025 年の AI の採用：個人と企業	10
6.1. 個人による利用	10
6.2. 企業での採用	10
7. コードの分析とメンテナンスに AI を適用する	12
7.1. 開発者にとっての「生の」AI の限界	12
7.2. ツールを備えたフレームワークが必要な理由	12
7.3. Visual Expert などのツールへの AI 統合の例	14
8. 最初のソフトウェア保守エージェントに向けて	15
8.1. コードレビューアシスタントから行動可能なエージェントへ	15
8.2. 人間の監督と安全装置の役割	15
9. 開発者、アーキテクト、運用担当者向けの実践的な推奨事項	16
10. 結論：共同知能と競争優位性	17
FAQ	18
参考文献	20

1. エグゼクティブサマリー

AI は、コード生成、レビュー、影響分析、文書化、デバッグ支援など、複雑なタスクにおいて開発者を支援するのに十分な成熟度に達しています。しかし、自律的な知能にはまだ程遠く、時には自信を持って誤りを犯すこともあります。

一方、その普及は加速しており、約 10 億人が定期的に利用しており、その一部は業務目的での利用が増加しています。78%の企業が AI を利用していると回答していますが、その可能性を十分に活用していると考えている企業はわずか 1%に留まっています。

このホワイトペーパーでは、開発ツールにおける AI の最新技術、その強みと限界、セキュリティとコンプライアンスの課題について概説し、コードの分析およびメンテナンスのワークフローに AI を統合する方法を示します。これは、高度な静的解析と推論 AI を組み合わせて複雑なコードの説明、コメント、改善を行う Visual [Expert](#) **での** 当社の経験に基づいています。

当社の仮説は以下の通りです。

- AI は短期的あるいは中期的には開発者やアーキテクトに**取って代わることはありませんが**、
- しかし、適切なガバナンスと体系的な人的監督のもと、管理された環境で使用されることを条件として、競争力を維持するために AI はすぐに**不可欠なもの**となるでしょう。

2. 背景：開発チームにとって AI が不可欠になる理由

これまでと同様に、開発チームはますますプレッシャーに直面しています。

- リリースサイクルの短縮化に伴う、より迅速な成果の提出
- ますます大規模化・重要化する既存アプリケーションの維持
- 新しいテクノロジー（クラウド、マイクロサービス、API、セキュリティ、AI 自体）を統合すること
- ますます厳格化するコンプライアンスおよびサイバーセキュリティの制約を順守すること

このような状況において、AI は生産性を高めるための潜在的な手段です。

しかし、汎用的なチャットボットを使用するだけでは不十分です。構造化されたコンテキストがなく、システム全体を理解していない場合、AI は部分的に盲目になります。つまり、もっともらしいが誤った回答を提案したり、実際の依存関係を無視したり、重要な制約を軽視したりする可能性があるのです。

したがって、課題は単に AI を使用することではなく、AI が真に価値をもたらすことができるタスクを特定し、豊富で構造化されたコンテキストを提供できる既存の開発ツールに AI を統合することです。

3. 2025 年の AI の現状

3.1. ANI、AGI、ASI：現在の現実

文献では、AI の進化における 3 つの大きな理論的段階が区別されている。

1. ANI – 人工狭義知能

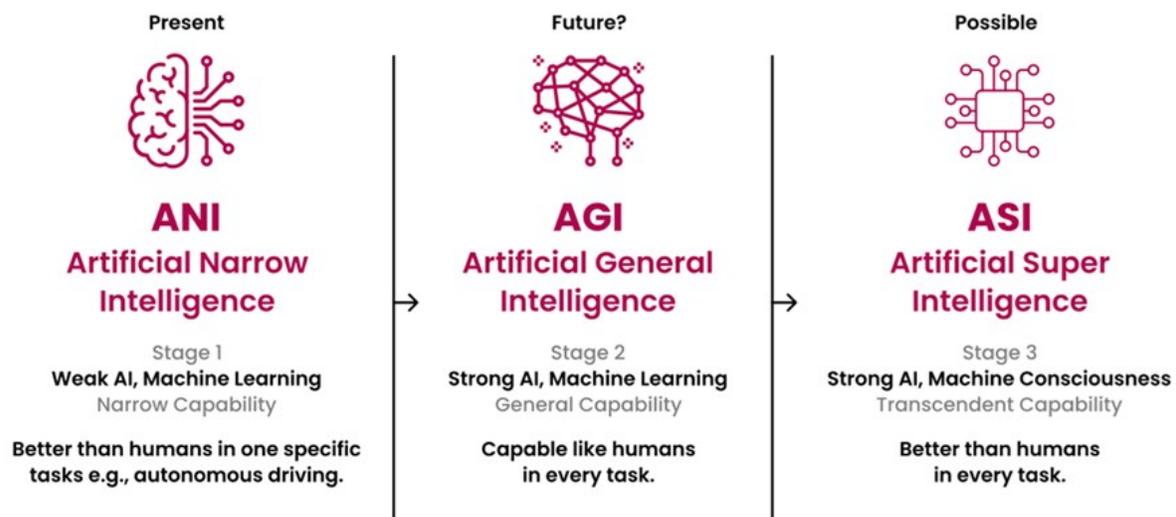
これは今日の AI、つまりチャットボット、画像認識、診断支援、コードアシスタントなどです。特定のタスクに特化しており、全体的な理解力はありません。明確に定義された枠組みでは効果的ですが、その枠組みの外では機能しません。

2. AGI – 汎用人工知能

人間と同等の柔軟性（抽象的思考、創造性、世界理解）を持ち、新しい状況に適応し学習できる AI。現在のモデルではこれに近づくものはなく、専門家たちはこの技術が中期的にこの段階に到達することには懐疑的です。

3. ASI – 人工超知能

この AI は、あらゆる分野（科学、戦略、直感など）で人間を凌駕し、自らのモデルを改善することができる。これは、むしろサイエンスフィクションの領域である。この段階にいつか到達すると主張する産業関係者はいない。



実際には、今後数年間は、非常に強力ではあるものの、そのトレーニングやコンテキストに依存する特殊なモデルである **ANI** の領域にとどまることになるでしょう。

3.2. 単純な会話からエージェントへ：会話型、生成型、推論型、エージェント型 AI

2022 年以降、ANI モデルは段階的に進歩しています。

1. 会話型 AI（2022 年～2023 年）

- 自然言語理解。
- 質疑応答、説明、テキストの言い換え。
- 限界：静的な回答、記憶容量が少ない、頻繁なエラー。

2. 生成型 AI (2023 年～2024 年)

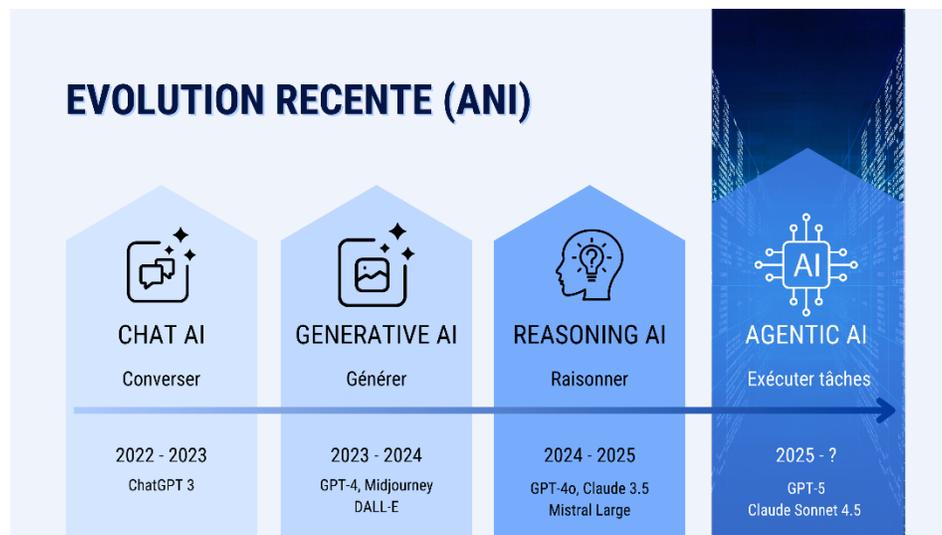
- テキスト、画像、コード、音楽、動画の生成 (GPT-4、DALL-E など)。
- スタイル、トーン、構造に従うことができる。
- 限界：意味の理解が表面的、推論能力に制限がある。

3. 推論 AI (2024 年～2025 年)

- 推論の段階を連続的に実行し、部分的に修正を行い、ツール (検索、計算など) を使用できる。
- 開発ツールにとっての主な利点：問題解決、デバッグ、コードの説明、構造化されたレビュー。
- 限界：「局所的な」論理はしばしば優れているが、全体像の把握は弱い。依然として自律性がなく、常に誤りを犯す。

4. Agentic AI (2025 - …)

- AI が目標達成のために一連のタスクを **計画・実行**。
- 推論と行動 (ナビゲーション、API の呼び出しなど) を組み合わせる。
- 限界：依然として人間の指示に依存し、常にミスを犯す。



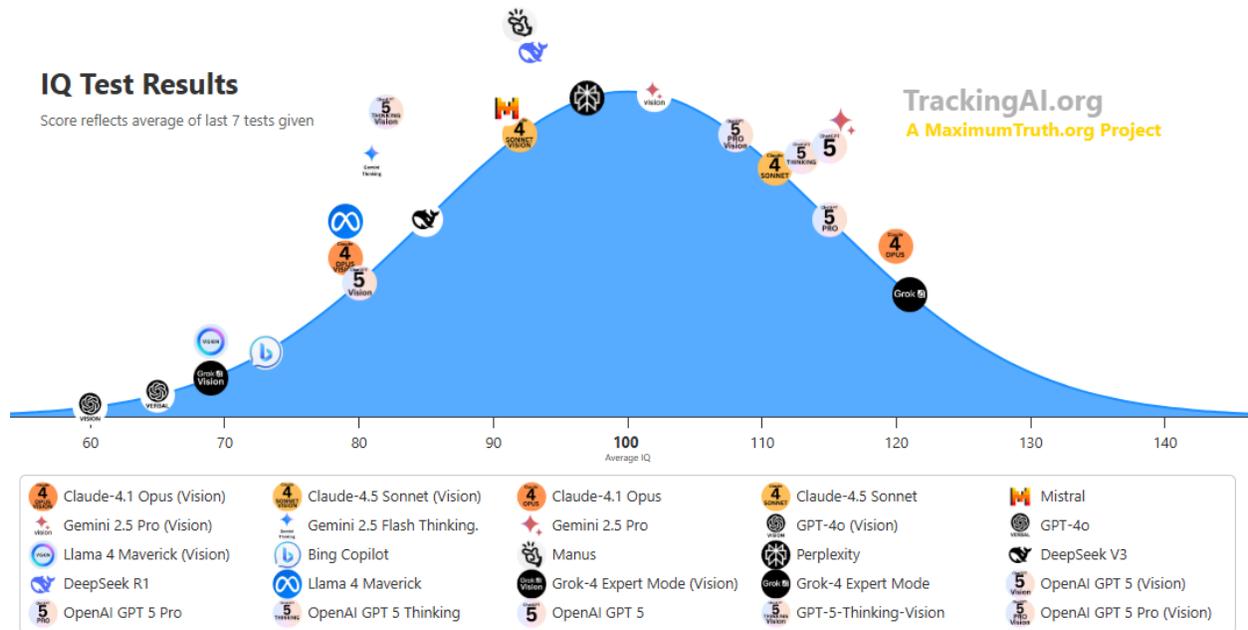
Visual Expert の場合、現時点では未成熟とみなしているエージェントにイニシアチブを委ねるのではなく、構造化された手続き的フレームワークに組み込まれた **推論 AI** を使用しています。

これにより、AI は、コードの静的解析後に Visual Expert が提供する特定のコンテキストに基づいて、開発者が適切なタイミングで起動することで、コードの説明、コメント、分析、改善に使用されます。

4. 開発タスクにおける AI の真の実力と限界

4.1. 純粋なパフォーマンス：IQ テストとベンチマーク

最近のベンチマークでは、一部のモデルが標準化された IQ テストにおいて人間と同等のパフォーマンスを達成していることが示されています。したがって、理論的には、AI は特定の点において人間と「同等」であると言えます。



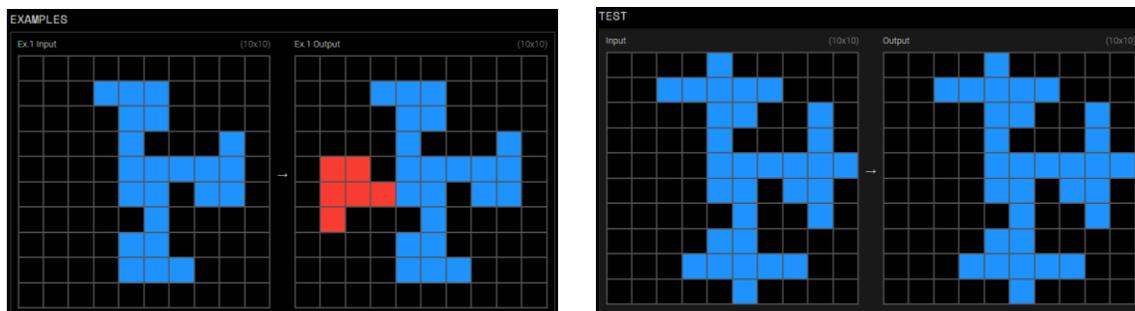
2025 年 10 月に Tracking.org が市場主要モデルに対して実施した IQ テストの結果

2025 年には、最高のモデルが象徴的な 100 のしきい値を超えました

…しかし、これらの数値は主に、明確に定義されたタスク（形式的な問題解決、論理、文章理解）における**純粋な処理能力**を測定したものです。堅牢性、適応性、企業コードベースのような複雑で生きたシステムを扱う能力は反映されていません。

4.2. 適応性テスト：AI がまだ失敗している分野

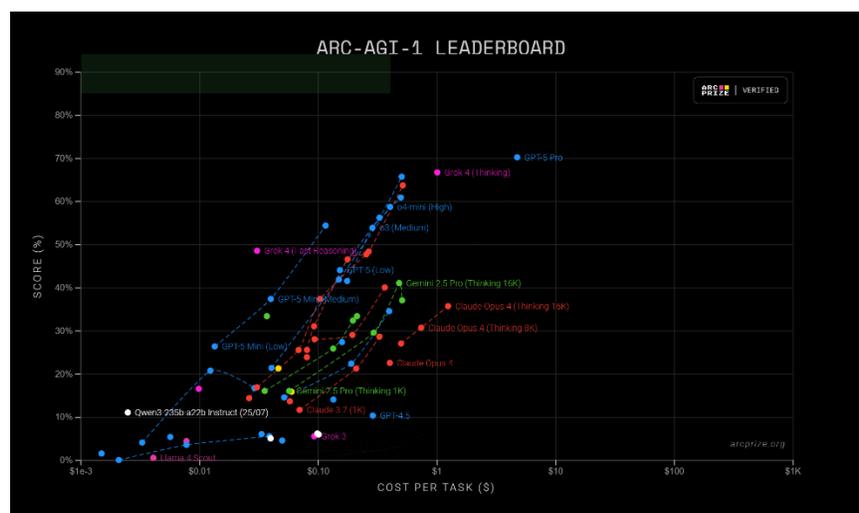
AI の限界をよりよく評価するために、[ARCPrize](#) 財団などの組織は、暗黙のルールや変化する状況の中で、新しいタスクへの適応能力を測定するテストを設計しています。



[ARCPrize](#) が提案するテストの例 - 左側が例、右側が同様のテスト問題。これらのテストは、AI が訓練を受けていないタスクを課すように設計されています。

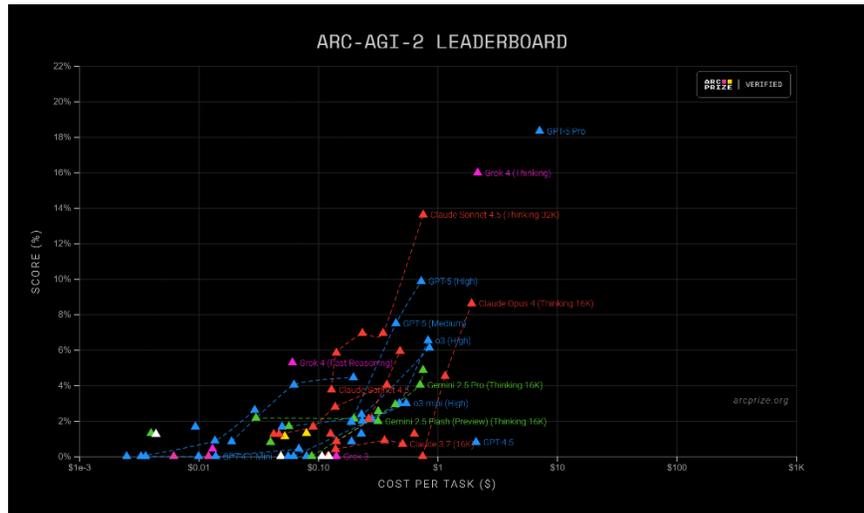
結果：

- ほとんどモデルは、人間が簡単に解けるパズルで **60%以上の正解率**を超えるのに苦労しています。



[ArcPrize](#) は、[この最初のテストシリーズを作成し](#)、「[基本的な流動的知能](#)」
- AI にとって難題となるパズル形式の単一タスクで測定するために作成されました。

- 適応性と効率性を組み合わせたテストでは、**大半のモデルが 10% 未満**、最高のモデルでも 20% に達していません。



ArcPrize はその後、第 2 弾のテストを開発しました。これは「システムに、高い適応性と効率性の両方を実証するよう要求する」ものです。結果から、AI がその純粋な能力にかかわらず、これらのテストを成功させるのがいかに難しいかがわかります。

- これらのスコアは、従来のベンチマークで測定される純粋な処理能力よりもはるかに遅いペースで向上しています。

つまり、AI は「これまでに見たことがある」形式の明確に定義されたタスクを与えられた場合には**非常に強力**ですが、以下の状況では脆弱なままです。

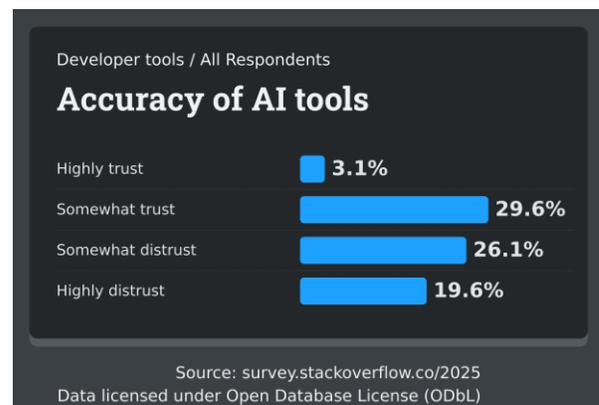
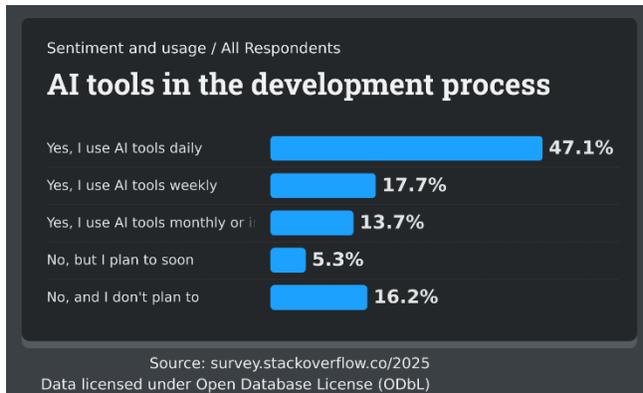
- 新しい状況から一般化すること
- 矛盾する制約の間で判断を下す
- 数百万行のコードからなる業務アプリケーションのような複雑なシステムを俯瞰的に捉える

4.3. 信頼性、幻覚、共同知能の必要性

これらの限界は、開発者の日常業務に直接影響します。

- AI は**自身の信頼性を評価することができません**。一見一貫性のある推論に基づいて、自信を持って誤った回答を生成する場合があります。
- モデルは**文脈に非常に敏感**です。明確な指示や関連データがない場合、曖昧な回答をしたり、単にユーザーを「喜ばせる」だけにとどまったりします。

たとえば、[Stack Overflow Developer Survey 2025](#) によると、開発者の 84% はすでに AI ツールを使用していますが、その 46% は生成された回答を信頼しておらず、ほぼ半数が、これらのツールは複雑なタスクの処理に苦労していると考えています。



使用と信頼のギャップ：開発者の 84% が AI ツールを使用しているが、46% がその結果を信頼していない。(出典：StackOverflow)

ここで**共同知能**の利点が明らかになります。

- AI が結果を生成し、人間がそれを検証する、
- あるいは、人間が結果を生み出し、AI が検証と改善を支援する。

ソフトウェアメンテナンスの分野では、多くの場合、2 番目のアプローチが最も適切です。AI は**レビューと分析のアシスタント**として機能しますが、最終的な決定の責任は開発者が負います。

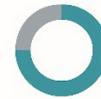
5. セキュリティ、コンプライアンス、ガバナンス：AI を導入するための前提条件

企業における AI 導入の主な障壁は、現在、セキュリティとコンプライアンスに関連しています：秘密情報の漏洩、個人データ、業界規制、責任などです。

2024 年から 2025 年にかけての主なトレンドは以下の通りです。

- 企業は **AI ガバナンス** 方針を導入しています：
 - 59% はすでに AI の使用および管理に関する規則を導入している
(出典：[Pacific](#))
 - 98% が AI ガバナンスの予算増額を予定
(出典：[OneTrust](#))。
- 欧州は、AI の使用を規制し、重要な AI システムへの信頼を強化することを目的とした **AI 法** を採択しました。
- 主要な有料モデルプロバイダー（OpenAI、Claude、Gemini など）はすべて、**データの機密性を保証する契約オプション**を提供しています。
 - 顧客データをモデルの再トレーニングに使用しない
 - 環境の分離
 - ログの制限と管理

Gouvernance IA en entreprise (enquête 2025)



59%
gouvernance IA
en place



98%
budget gouvernance
augmenté

2025 年には、大多数の企業が AI ガバナンスを導入する。

Visual Expert の場合、これらの保証は、モデルトレーニングや提供サービスに直接関係のないその他の用途での顧客データへのアクセスを明示的に禁止する **DPA（データ処理契約）**によってさらに強化されています。

DevOps チームにとって、これは、以下の条件を満たせば、コードやビジネスデータを追加のリスクに **さらすことなく**、AI を開発ワークフローに統合できることを意味します。

- 規制上の制約（GDPR、AI 法など）に準拠したプロバイダーを選択すること。
- AI に送信されるデータの種類を明確に定義する。
- これらの要素をセキュリティポリシーおよびアーキテクチャレビューに組み込むこと。

6. 2025 年の AI の採用：個人と企業

6.1. 個人による利用

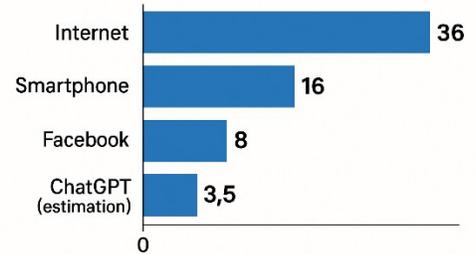
AI の個人利用は大幅に増加しています。

- 定期的なユーザー数は約 **10 億人** と推定され、そのうち 8 億 5000 万人は OpenAI を利用している。
- 米国では、**人口の 62%** が少なくとも週に 1 回は AI を利用している（出典：[Pew Research](#)）。
- ユーザーの **19% から 27%** が、仕事で使用していると回答しています（出典：[Gallup](#)）。

多くの開発者が現在、AI を以下の目的で使用しています。

- 単体テストを書く；
- アルゴリズムについて簡単な説明を求める
- 関数のスケルトンやインフラスクリプトを生成する；
- チケットやドキュメントの書き直し

Nombre d'années pour atteindre un milliard d'utilisateurs réguliers

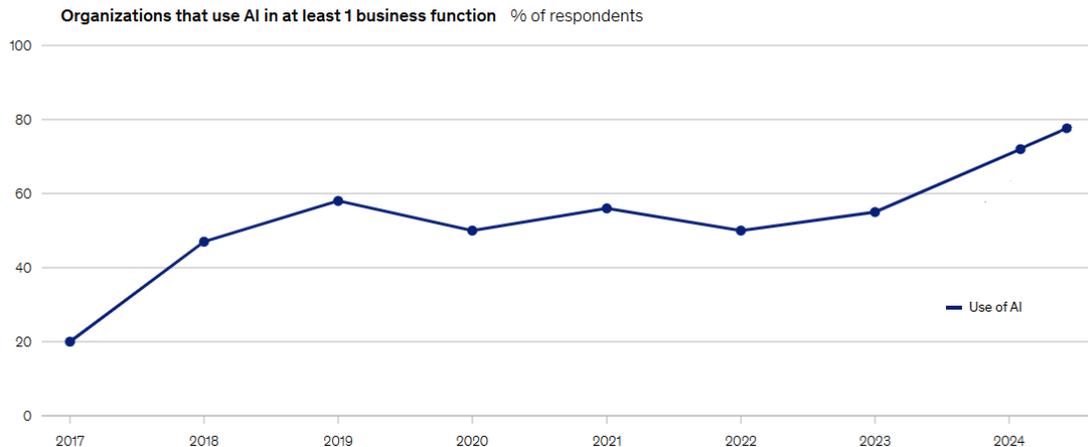


ChatGPT は、Facebook の 2 倍、スマートフォンの 4 倍、インターネットの 9 倍のスピードでユーザーを獲得しています

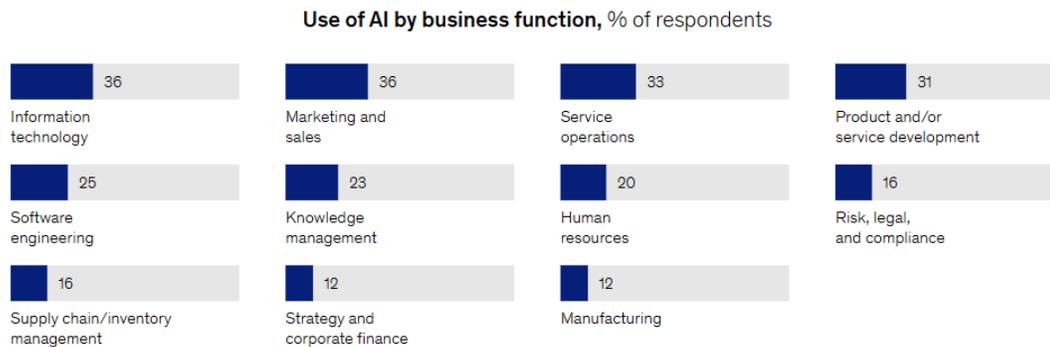
6.2. 企業での採用

企業側では、逆説的な状況が見られます。

- **78% の企業が**、少なくとも 1 つの機能で AI をすでに活用しており、



- うち **36%**は IT 分野 (出典: [McKinsey](#))。



- しかし、それを「十分に」活用しているとはわずか **1%**でした (出典: [ピュー・リサーチ](#))。
- アメリカの従業員のうち、AI に関する**正式な研修**を受けたのはわずか **12%** でした (出典: [ピュー・リサーチ](#))。
- それにもかかわらず、**17%**の企業が AI によって少なくとも 5%の EBIT の増加を確認している (出典: [McKinsey](#))。

開発チームにとって得られる教訓は明らかです。

- AI は、まだ組織化が進んでいない組織でも、**すでに価値を生み出している**。
- 最大の利益は、用途の**工業化**、トレーニング、既存のツール (IDE、CI/CD パイプライン、コード分析ツール) への統合によってもたらされる。

7. コードの分析とメンテナンスに AI を適用する

7.1. 開発者にとっての「生の」AI の限界

ChatGPT タイプのモデルに、単純なコピー&ペーストで「この機能を説明してください」や「この変更の影響を教えてください」と依頼するだけでは不十分です。

その限界はすぐに明らかになります。

- 不完全なコンテキスト：AI はコードの一部を見るが、依存関係、データモデル、構成上の制約などを認識できない。
- 部分的な視野：断片的な情報から全体像を把握することは不可能です。
- サイレントエラーのリスク：AI は、特定のケース、テーブル、トリガーなどを認識できず、情報不足を報告することはありません。

複雑で重要なシステムの場合、**運用上のリスクがあります**。AI のみに依存したリファクタリングは、機能を破壊したり、アプリケーションをブロックしたり、パフォーマンスを低下させたりする可能性があります。

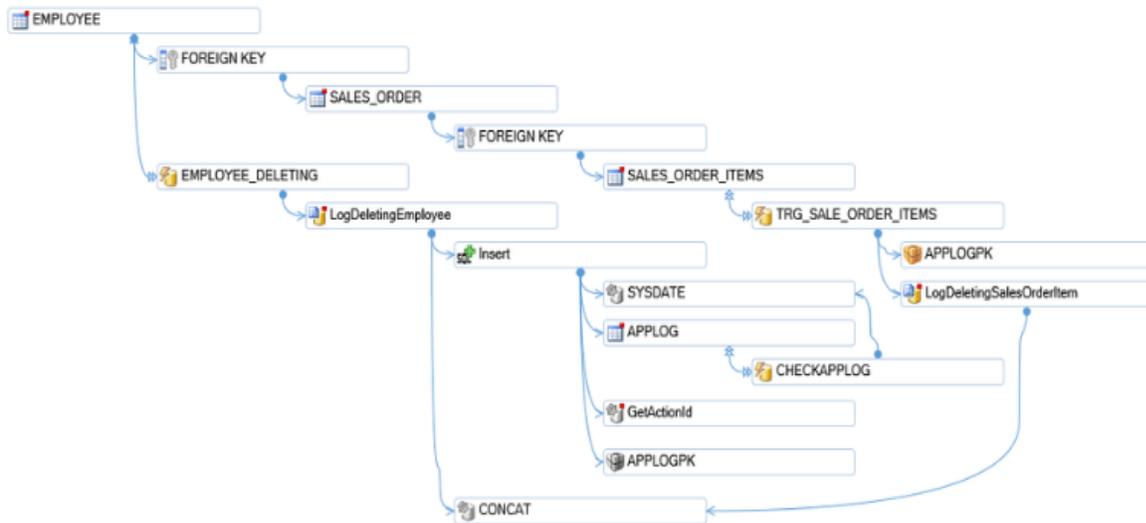
7.2. ツールを備えたフレームワークが必要な理由

アプリケーションのメンテナンスにおいて生産的に活用するためには、AI には以下が必要です：

1. 完全かつ構造化されたコンテキスト

- [オブジェクトのマッピング](#) (プロシージャ、トリガー、テーブル、UI コンポーネント、サービスなど)

- 依存関係と呼び出しパスの分析 ([相互参照](#)、呼び出しグラフなど)
- アプリケーション、ボリューム、パフォーマンスに関するメタデータ



Visual Expert は、オブジェクト、機能、データ間の呼び出しチェーン全体をマッピングします。
この構造化されたコンテキストにより、AI は Oracle、SQL Server、PowerBuilder、その他のコードに関する関連情報を提供することができます。

2. 構造化され、管理された質問（入力）

- 明確な目的：説明、コメント、影響の評価、リスクの検出
- 期待される回答の構造（リスト、表、要約 + 詳細など）
- 明示的な制約（創作しない、不足情報を報告する、確認する）。

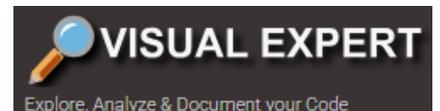
3. 既存ツールへの統合

- コード分析ツールまたは IDE から
- ログ、トレーサビリティ、ピアレビューの可能性付き。

これらの制約を満たすため、[Visual Expert](#) は静的コード解析を使用して、AI にアプリケーションに関する構造化データを提供し、AI はそのデータを利用して、文脈に応じた検証可能な回答を生成します。

7.3. Visual Expert などのツールへの AI 統合の例

Visual Expert では、AI は事前定義された機能に組み込まれています。



- オブジェクトまたはプロシージャのビジネスロジックの説明（例）
- 既存のコードに[コメント](#)を追加する
- [コードの検査](#)で特定された欠陥に対する[解決策を提案・説明](#)
- [パフォーマンス分析](#)で特定された遅いクエリや関数を[最適化](#)します。

```
<> Source code
50 open cur for Prod_name using temp_1, temp_2;
51 FETCH cur INTO prod_1, prod_2; -- Replacing loop with cursor.
52 exit when cur%NOTFOUND; -- Exit when no more data is available
53 if instr(prod_1, '.') = 0
54 then prod_1 := temp_1 || '.' || cust_name;
55 end if;
56
57 If temp_1 is null Or temp_1 = '' Then
58     temp_1:=CUSTOMERS.GetFullname(temp_1, temp_2);
59     sp_deleteemployee(temp_1);
60 end if;
61
62 CLOSE cur; -- Closing cursor after use to free resources.
63 End If;
```

VISUAL EXPERT AI は、ORACLE PL/SQL において、非効率な LOOP 構文をカーソルベースのアプローチに置き換えることを提案し、パフォーマンスと保守性を向上させます

各機能について：

- Visual Expert は自動的にコンテキスト（コード、依存関係、メタデータ）を準備します。
- AI モデル向けに構造化されたクエリをフォーマットします。
- 開発者が確認、補完、修正できる応答を取得します。

これは共同知能の枠組みに則っています。AI は開発者に代わって行動するのではなく、より質が高く、より迅速に取得でき、アプリケーションの実際のアーキテクチャに沿った出発点を提供します。

8. 最初のソフトウェア保守エージェントに向けて

8.1. コードレビューアシスタントから行動可能なエージェントへ

現在の機能は、**最初のソフトウェア保守エージェント**を予見しています。

- 修正を**提案**したり、**包括的なリファクタリング**を準備したりできる。
- 特定の仮定（例えば、変更が**保守性やセキュリティの制約**を満たしているかどうか）を**確認する**
- 最終的には、**CI/CD のコンテキストにおいて、ビルド実現**プロセスの複数の段階、特に**継続的なコード検証**を**自動化すること**。

8.2. 人間の監督と安全装置の役割

しかし、短期的には、AI の限界により、いくつかの安全対策が必要となります。

- 本番環境での自律的な実行は不可
- 開発者またはアーキテクトによる体系的なレビュー
- 手動または自動化されたテスト（単体テスト、統合テスト、性能テスト）による補完/検証。

その目的は、メンテナンスを AI に「委任」することではなく、付加価値の低い作業（説明、文書化、反復的な分析）から人的リソースを解放し、アーキテクチャの決定や業務上の判断に専門知識を集中させることです。

9. 開発者、アーキテクト、運用担当者向けの実践的な推奨事項

1. 最初から適切なユースケースをターゲットにする

- 既存のコードの説明と文書化。
- 特定されたコードの問題に対する修正の提案。
- 開発者が検証しやすい、定義された範囲での最適化の提案（例：SQL クエリ）。
- その他？

2. コンテキストを提供するツールを活用する

- 既存のツールに統合されたソリューションを、「裸の」チャットボットよりも優先する。

3. ガバナンスの枠組みを定義する

- ソースコードおよび機密データに対する AI の使用方針。
- 準拠したサプライヤーの選択（AI 法、GDPR、DPA など）。

4. チームを育成する

- AI の限界（幻覚、文脈の必要性）についての認識を高める。
- ツールを備えた枠組みにおけるベストプラクティス。
- 既存のプロセスへの統合（コードレビュー、設計レビュー、事後検証）。

5. 6～18 か月のロードマップを策定

- フェーズ 1：限定的な範囲（1 つのアプリケーション、1 つのチーム）での制御された実験。
- フェーズ 2：成功したユースケースの一般化、ツールおよびパイプラインへの統合。既存プロセスの生産性向上
- フェーズ 3：AI と従来のツールの組み合わせの可能性を最大限に活用するためのプロセスの段階的な進化、および監督強化。

10. 結論：共同知能と競争優位性

AI は開発者、アーキテクト、運用担当者を置き換えるものではありません。しかし、AI をツールやプロセスに賢く統合できるチームは、メンテナンスサイクルの短縮、既存アプリケーションの理解の深化、特定のリスクの低減、大きな変化への適応力向上など、**大きな優位性を得ることができます。**

今日の課題は、AI が「十分に優れているか」ではなく、それを**どのように**活用するかです。

- 強固なガバナンス体制のもとで；
- 現在のツールやプロセスに統合することで
- 豊富なコンテキストを提供できるソリューションを活用し；
- 技術的な意思決定の中心に人間を据えること。

Visual Expert のような、**高度な静的解析と推論 AI** を組み合わせたソリューションが、その道筋を示しています。これらのソリューションは、AI を、信頼性が高く、生産性が高く、現代の情報システムのセキュリティおよびコンプライアンス要件に準拠した、**ソフトウェアメンテナンスのアシスタントに変えます。**

FAQ

1) AI は、既存のアプリケーションのメンテナンスに具体的にどのように役立つのか？

AI は、理解の促進や、特定の改善点の提案に特に効果的です。オブジェクトのロジックの説明、プロシージャの機能の概要説明、暗黙のビジネスルールの明確化、コメントの提案、最適化の提案などです。一方、依存関係を分析し、変更を安全に行うには、信頼性の高いマッピング（静的解析、呼び出しグラフ、コードとデータの依存関係）を提供するツールを利用する必要があります。AI はコードの一部しか把握していないため、すべての参照を「認識」することはできず、変更による副作用を報告することはできません。

2) AI は開発者、アーキテクト、運用担当者にとって代わるのか？

いいえ。AI は特定のタスクを支援または加速することはできますが、信頼性や全体像を保証するものではありません。最も効果的なアプローチは「共同知能」です。つまり、AI が提案し、人間が検証する、あるいは人間が実行し、AI が検証する、というものです。複雑なアプリケーションのメンテナンスに関しては、人間が引き続き操作の仲裁者であり、最終結果の責任者であり続けるでしょう。

3) コードや機密データを公開せずに AI を利用するには？

以下の要素を組み合わせることで実現できます：(1) 明確なガバナンス方針（送信可能/不可能なもの）、(2) 契約上の保証（DPA、非トレーニング）のあるサプライヤー [（例はこちら）](#)、(3) セキュリティ/アーキテクチャのレビュー。

4) Visual Expert の AI は、コードを貼り付けるチャットボットとどう違うのですか？

チャットボットは、ユーザーが与えたコードの部分しか認識できません。一方、Visual Expert は、コードとデータベース（データモデル、オブジェクト、依存関係、メソッド、属性など）を事前に分析することで、システム全体の構造を把握しています。その後、Visual Expert は AI を使用して、「人間的な」タスク、つまり、ビジネス [やロジックの説明](#)、[コメントの生成](#)、VE がコード内で発見した問題の解決のための修正や [最適化の提案など](#)、さらに一歩進んだ作業を行います。

5) Visual Expert の AI は、メンテナンスにおいて具体的にどのような役割を果たしているのですか？

主に以下の目的で使用されます。

- 説明：オブジェクトや手順などのロジックや「ビジネス」を説明します。
- [コメント](#)：可読性を高めるために、有用で（かつ一貫性のある）コメントを生成します。
- [修正](#)：[コードの自動検査](#)中に発見された問題に対する解決策を提案する。
- [最適化](#)：、Oracle PL/SQL、SQL Server TSQL、または PowerBuilder [のコード分析で特定されたパフォーマンスの問題](#)に対する改善や最適化を提案します。

6) VE は、これらの機能において AI のエラー（幻覚）をどのように制限しているのでしょうか？

使用範囲を限定する：

- AI は、Visual Expert が提供する、分析対象のコードに関連するすべての情報を含むコンテキストとともに、**特定のコード範囲**（プロシージャ、オブジェクトなど）で動作します。

- 詳細な指示（[Explain](#) / [Comment](#) / [Suggest a fix](#) / [Optimize](#)）による特定のタスクの実行のために呼び出され、自由なチャットではなく、
- その結果は、開発者が簡単に**確認および検証**できるよう、適切な形式で表示されます。

7) Visual Expert とその AI を継続的インテグレーション（DevOps）のワークフローに統合するには？

シンプルなモデル：

1. Visual Expert を使用して定期的な分析（毎日、各ビルドごとなど）を計画し、Oracle PL/SQL、SQL Server T-SQL、PowerBuilder などの [コードを自動的に](#) マッピングおよび [検証します](#)。
2. Visual Expert の AI を使用して、発見された問題（コードと関連する問題の説明、修正や最適化の提案など）を修正します。
3. マージ/リリース前に、**レビューとテスト**（単体テスト、統合テスト、性能テスト）を通じて体系的に検証します。

8) AI はどのようなメンテナンスタスクで生産的であり、どのようなタスクでは生産的ではないのか？

AI は、主に「説明的」で反復的なタスク（オブジェクトの理解、言い換え、コメントの生成、レビューの支援、テストの提案、ローカルな最適化のヒントなど）を高速化します。一方、システム全体の視野を必要とするタスク（依存関係の分析、リファクタリングなど）には、これらの弱点を補う専用ツールと統合されない限り、適していません。

Visual Expert について

Visual Expert は、PowerBuilder、Oracle PL/SQL、SQL Server T-SQL 向けのコード分析ソリューションです。静的解析（相互参照、呼び出しグラフ、影響分析、パフォーマンス診断）により、複雑なコードベースのマッピング、文書化、保守を実現します。

Visual Expert は、ビジネスロジックの説明、コメントの生成、修正の提案を行う推論 AI 機能を統合しています。AI は、静的解析によって提供される構造化されたコンテキストに基づいており、関連性の高い結果を保証します。

コードで AI を試してみませんか？

Visual Expert をダウンロードしてください。

参考文献

[Tracking AI](#) – AI モデルの IQ テスト

- 著者/会社：Maxim Lott、[Tracking AI](#)
- タイトル： *Tracking AI – IQ テスト*
- 日付：2024 年～2025 年

[ARCPrize](#) – ARC-AGI ベンチマーク

- 著者/会社：ARC Prize Foundation
- タイトル： *ARC-AGI – 汎用知能ベンチマーク*
- 日付：2025 年にコンテンツ更新

Stack Overflow – 開発者による AI の採用

- 著者/会社：Stack Overflow
- タイトル： [Stack Overflow Developer Survey 2025](#)
- 日付：2025 年

[ピュー・リサーチ・センター](#) – 一般大衆による AI の利用と認識

- 著者/会社：Colleen McClain、Brian Kennedy、Jeffrey Gottfried、Monica Anderson、Giancarlo Pasquini、[ピュー・リサーチ・センター](#)
- タイトル： *米国一般市民と AI 専門家による人工知能の見方（日常生活における人工知能：見解と経験のセクションを含む）* [ピュー・リサーチ・センター+1](#)
- 日付：2025 年 4 月 3 日（2024 年に収集したデータ）

[Gallup](#) – 職場における AI の利用

- 著者/会社：ライアン・ペンデル、ギャラップ
- タイトル：「職場での AI の利用は 2 年間でほぼ 2 倍に」 [Gallup.com](#)
- 日付：2025 年 6 月 15 日

マッキンゼー – 企業における AI の採用と経済的影響

- 著者/会社：マッキンゼー・アンド・カンパニー
- タイトル：「2024 年初頭の AI の現状：汎用 AI の導入が急増し、価値を生み出し始める」（および AI の現状に関する関連調査） [マッキンゼー・アンド・カンパニー+1](#)
- 日付：2024 年 5 月 30 日

Pacific AI – AI ガバナンスの実践の成熟度

- 著者/会社：David Talby、Pacific AI
- タイトル：「2025 年 AI ガバナンス調査が明らかにした、AI の野心と運用準備の間の重大なギャップ」 [Pacific AI](#)
- 日付：2025 年 6 月 17 日

OneTrust – AI ガバナンスの予算と優先事項

- 著者/会社：OneTrust
- タイトル：2025 年 AI 対応ガバナンスレポート [OneTrust](#)
- 日付：2025 年

欧州連合 – AI 法

- 著者/機関：欧州議会および欧州連合理事会
- タイトル：人工知能に関する調和規則を定める欧州議会および理事会規則 (EU) 2024/1689 (人工知能法) [EUR-Lex](#)
- 日付：2024 年 6 月 13 日 (2024 年 7 月 12 日に EU 官報 L 2024/1689 掲載)