生成AIとQAの関わりとそれぞれの実践的アプローチのご紹介

2026.2月版　公開用抜粋版

株式会社ベリサーブ

執行役員 研究開発部長
QAエンジニア

松木 晋祐

AIとQAの関わりの3つ＋２の軸

※赤枠が本日のスコープ

QA4AI

AI製品をQAする

AI4QA

QA活動にAIを
活用する

QA4AIDD

AI駆動開発の
QA

PMO4AIDD

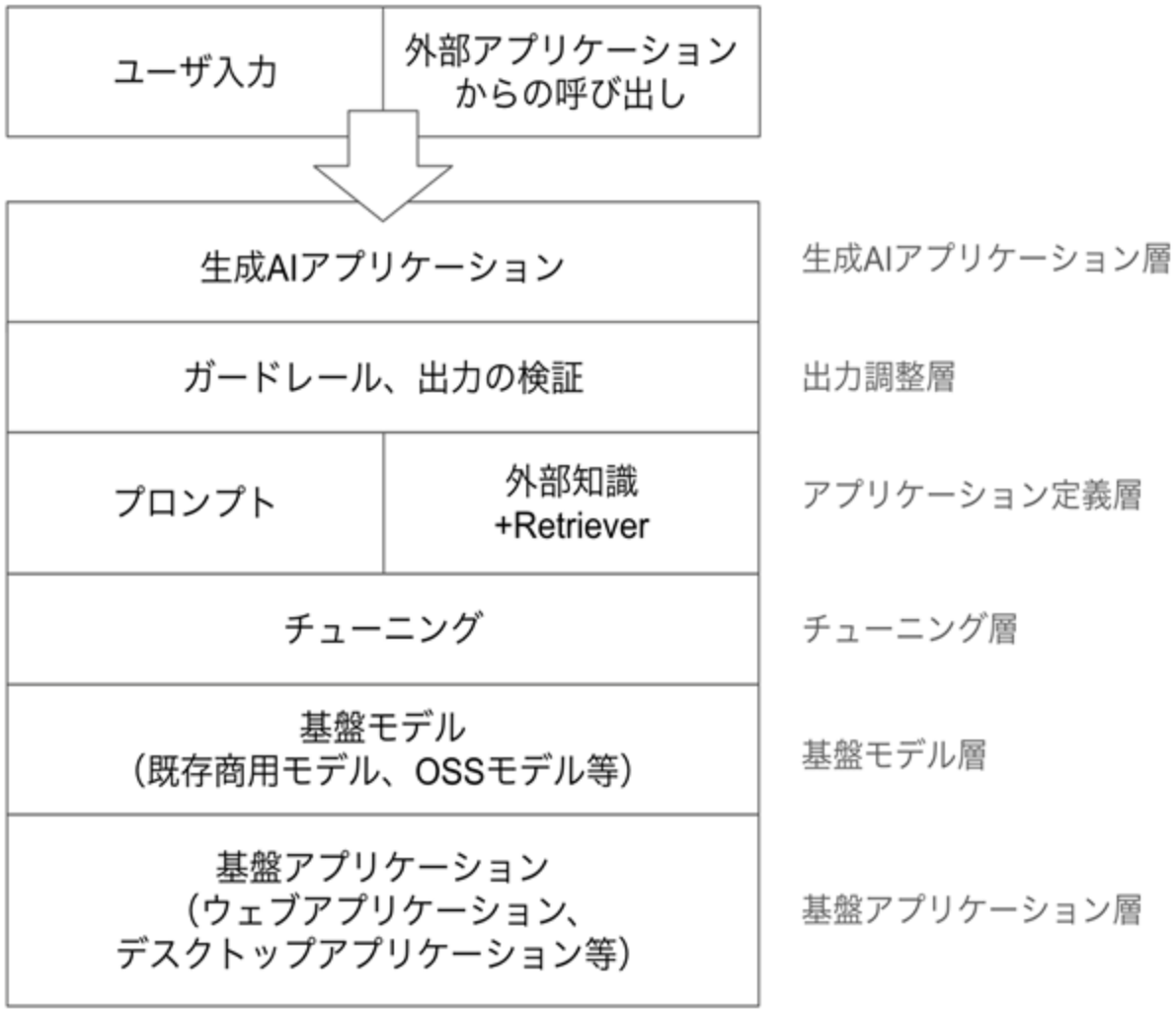
AIDD

AI駆動開発
の推進

QA4AI

- AIシステムの品質保証の概念について整理する
- テスト、評価戦略を考えるためにAIシステムの「つくり」を理解する
- 評価観点モデルを利用して実際の評価、メトリクスを設計する
- AIシステムのセキュリティテスト「レッドチーミング」

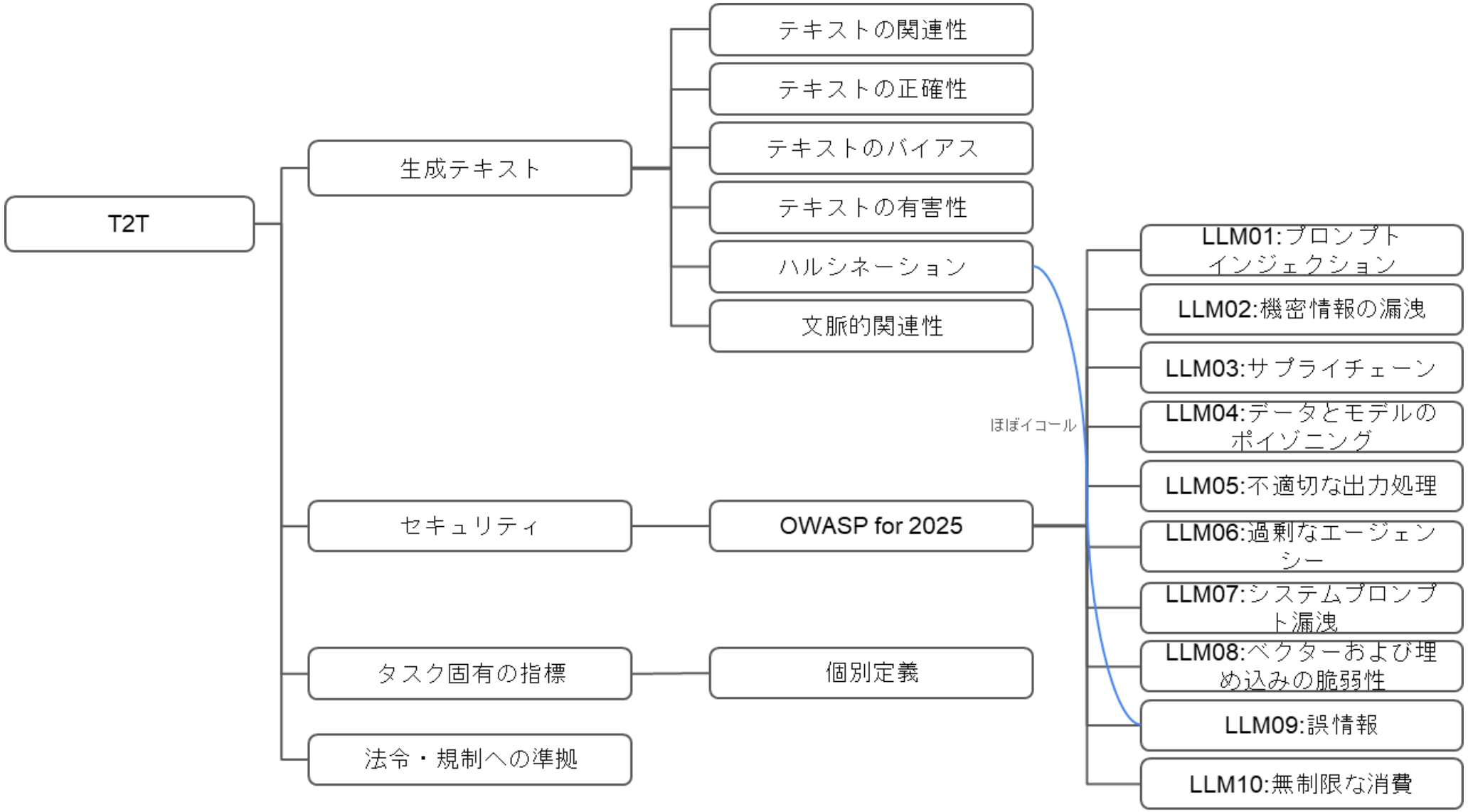
生成AIアプリケーションの基本的な構造モデル



- アプリケーションの構造を把握したうえで、QAが基本的に関わるのは「生成AIアプリケーション層」の振る舞いとモニタリング
- テスト観点モデルに基づいた観点の抽出と、個別タスクに対する観点定義とメトリックの設計（評価方式、閾値）
- そのメトリックの継続的な計測のための基盤の運用
- オフライン評価とオンライン評価、後者はモニタリング

出典：
「生成AIアプリケーションのテスト」（仮）

生成AIアプリケーションの評価観点基盤モデル：T2T



生成AIアプリケーションの評価観点基盤モデル：T2T

- 生成AIアプリケーション（生成AIが振る舞いのコアにある製品またはフィーチャー）のテストは、その動作原理の違いから、これまでの演繹的なソフトウェア開発によって作られた製品のテストと根本的に異なる＝「評価」が必

1.1.1 確率的出力とその影響

生成AIは、大量のテキストデータや画像データなどを学習し、統計的な手法で出力を生成します。そのため、同じ入力に対しても必ずしも同一の結果が返されるわけではありません。たとえば、チャットボットに同一の質問を行った場合、微妙な表現の違いや語順の変化、場合によっては意味合いが若干変化した回答が返されることがあります。この確率的出力は、利用者にとって柔軟性や創造性をもたらす一方、品質の一貫性や再現性の面では課題となります。テスト工程においては、こうした不確実性を統計的手法やサンプリングを用いて評価し、平均的な動作や異常値の有無を検証する必要があります。

1.1.2 ブラックボックス的な内部処理

生成AIの内部は、従来のプログラムのように明確なロジックの積み重ねではなく、ニューラルネットワーク内部の重み付けや学習パターンによって結果が導かれます。このブラックボックス的な性質は、原因不明の不具合や予期せぬ出力を生じやすく、従来の入力ー出力の単純な対応関係でのテスト手法では十分に捉えられない問題を含んでいます。たとえば、生成されるコンテンツにおいて誤情報や不適切な表現が含まれてしまった場合、その原因の追及が極めて困難となるため、内部ロジックの透明性向上とともに、出力の質を多角的に評価する仕組みが求められます。

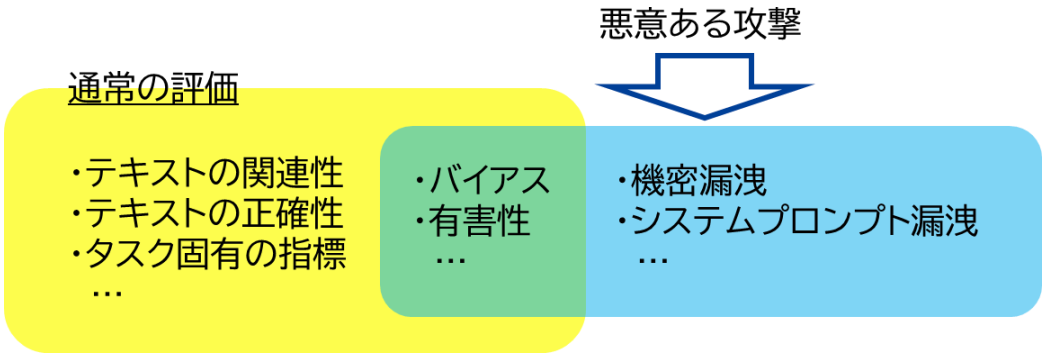
1.1.3 文脈依存性とダイナミックな挙動

生成AIは、過去の対話履歴や外部環境、さらにはユーザーごとの属性情報など、様々な文脈情報をもとに出力を生成します。この文脈依存性は、システムが常に同一の初期状態から出発しているわけではないことを意味し、状況に応じた柔軟な対応が求められます。たとえば、同じ質問でも、利用者の以前の対話内容や外部ニュース、季節やイベントなどの影響で、回答の内容やニュアンスが変化することがあります。このような動的な挙動に対応するためには、テストプロセスも動的なシナリオを想定した評価が必要となり、単一の静的なテストケースだけではなく、リアルタイムでのフィードバックループや継続的な評価体制が必須です。

出典：
「生成AIアプリケーションの評価」（仮）

生成AIアプリケーションのテスト：レッドチーミング

- 生成AIアプリケーションのセキュリティテストは、非生成AIアプリケーションのセキュリティテストと趣が異なる



- 📏評価項目の例
  - セキュリティ：システムプロンプトや内部コード漏洩の有無
  - 法規制対応　：景品表示法・薬機法などの法令準拠チェック
  - 幻覚・誤情報検出：事実誤認や不正確情報の検出
  - 有害性/バイアス　：性別・宗教・政治などの偏向表現、有害言語
- 🎭攻撃手法の例
  - Prompt Injection　：入力に巧妙な命令を埋め込み出力を誘導
  - Roleplay攻撃：特定キャラクターを演じさせポリシー回避
  - Gray Box Attack　：部分的なシステム知識を利用した攻撃
  - Goal Redirection　：目的をすり替えて不正な出力を引き出す
  - Input Bypass　：運用上の緊急性を装い制約を回避
  - Linear Jailbreaking　：会話を行いながら段階的に説得

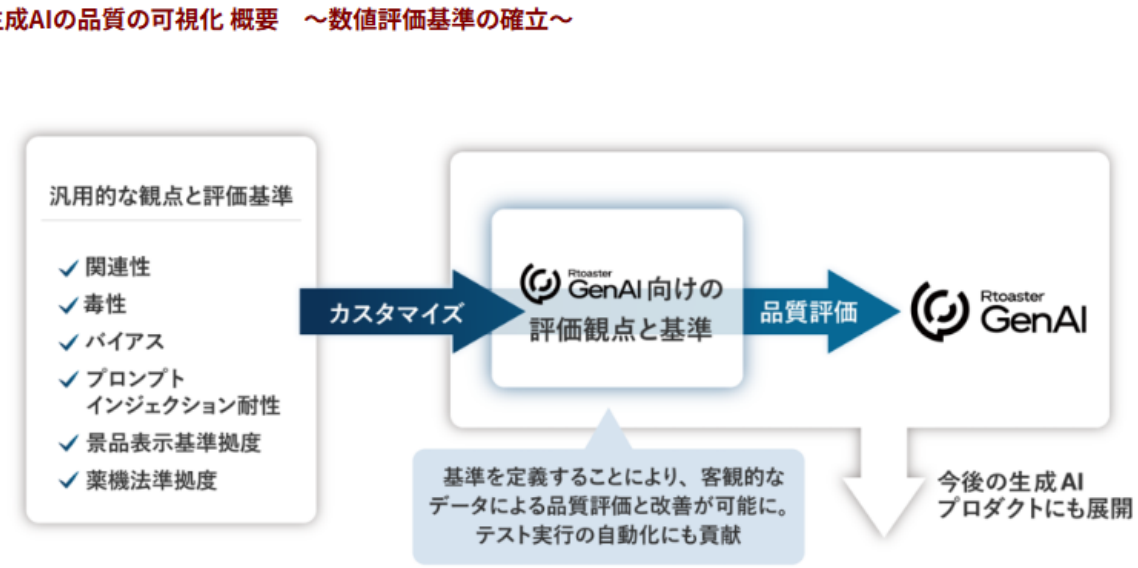
サービスデリバリ事例：ブレインパッド様

Rtoaster GenAI

BrainPad × VERISERVE

本発表のポイント

- 企業におけるデータ・AI活用に20年以上携わってきたブレインパッドと、ソフトウェア検証業界で累計41,000件以上のプロジェクトに携わってきたベリサーブの2社が協力。
- 実際に提供中の生成AIプロダクト「Rtoaster GenAI」を対象に、実践的・具体的な方法論を構築。
- AIの幻覚「ハルシネーション」や、倫理上の不適切さなどの品質の数値化・可視化を実現。回答内容を"ものさし"で測定し、プロダクトの改善サイクルを高速化。

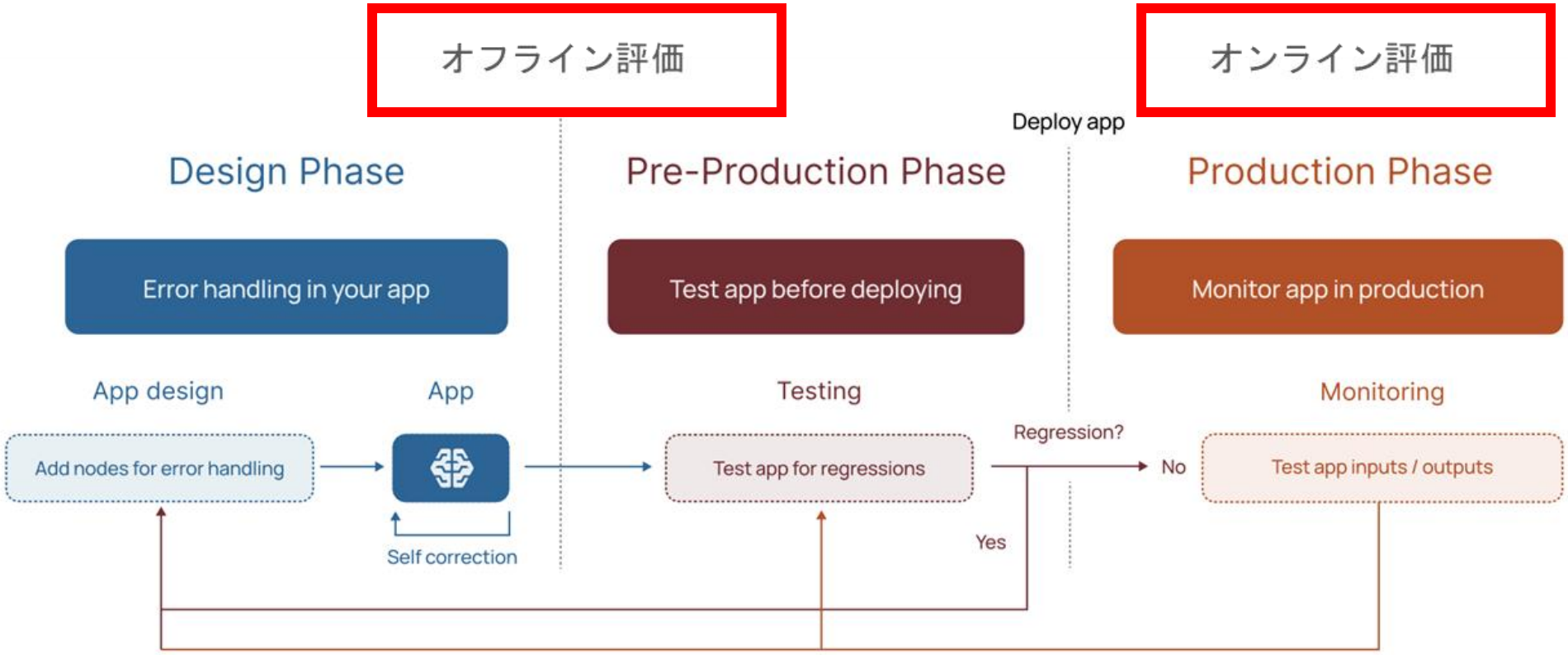


QA4生成AIアプリケーションによる、Rtoaster GenAIの品質基準の策定と今後の展望

先進AI搭載で市場をリードする
レコメンドエンジンのさらなる品質向上に貢献

テストアプローチと役割分担の例

キーになるのは「フライホイールの形成」

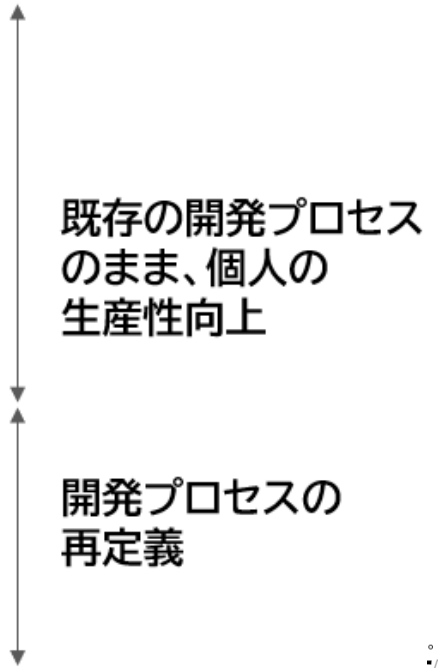


プリプロダクション（リリース前）の自動評価環境を、プロダクション（本番）環境でも
同様に動作させて、**実データでも評価を行い続ける**ことが重要！

QA4AIDD：種類と対応

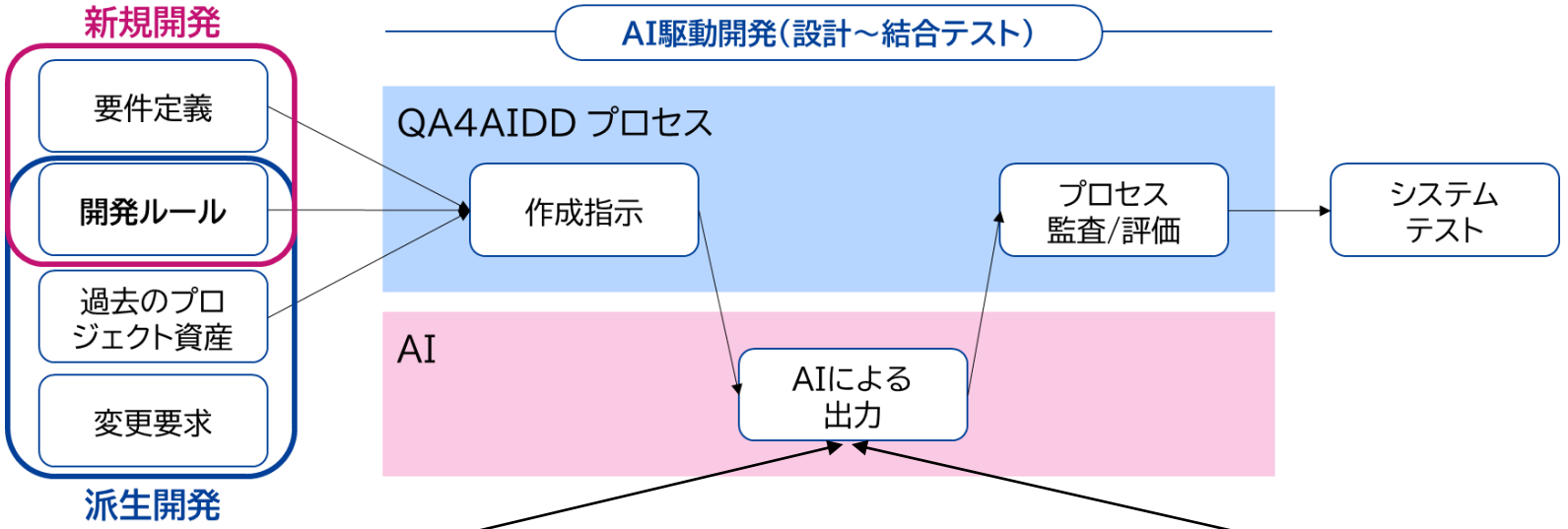
- VibeCoding までの間は、既存の開発プロセスと変わらないため、
  成果物＝プロダクトのQAに留まる
- SDDから、「AI駆動開発」のプロセスQAを検討する必要性が出てくる

種類	特徴
チャットスニペット	自然言語で指示 → コード断片を即生成
VibeCoding	AIと対話しながら、実装の方向性やスタイルを模索していく「対話型プロトタイピング」
コパイロット型（ペアプログラミング支援）	GitHub Copilot のように、IDEに組み込まれてリアルタイムに補完・提案してくれる方式
仕様駆動開発（Specification-driven Development）	自然言語の要件や仕様からコード・テスト・ドキュメントを自動生成
自律型エージェント開発	AIがタスクを分解 → 設計 → 実装 → テストを自律的に繰り返す



QA4AIDD：プロセス統制＋プロセスQA

- 「プロセスの統制」を担うQA4AIDDに加えて、既に指摘されている
  セキュリティ、内部品質への対応をAI駆動開発プロセスに織り込む必要がある



セキュリティ
(脆弱性を含むコードへの対策)

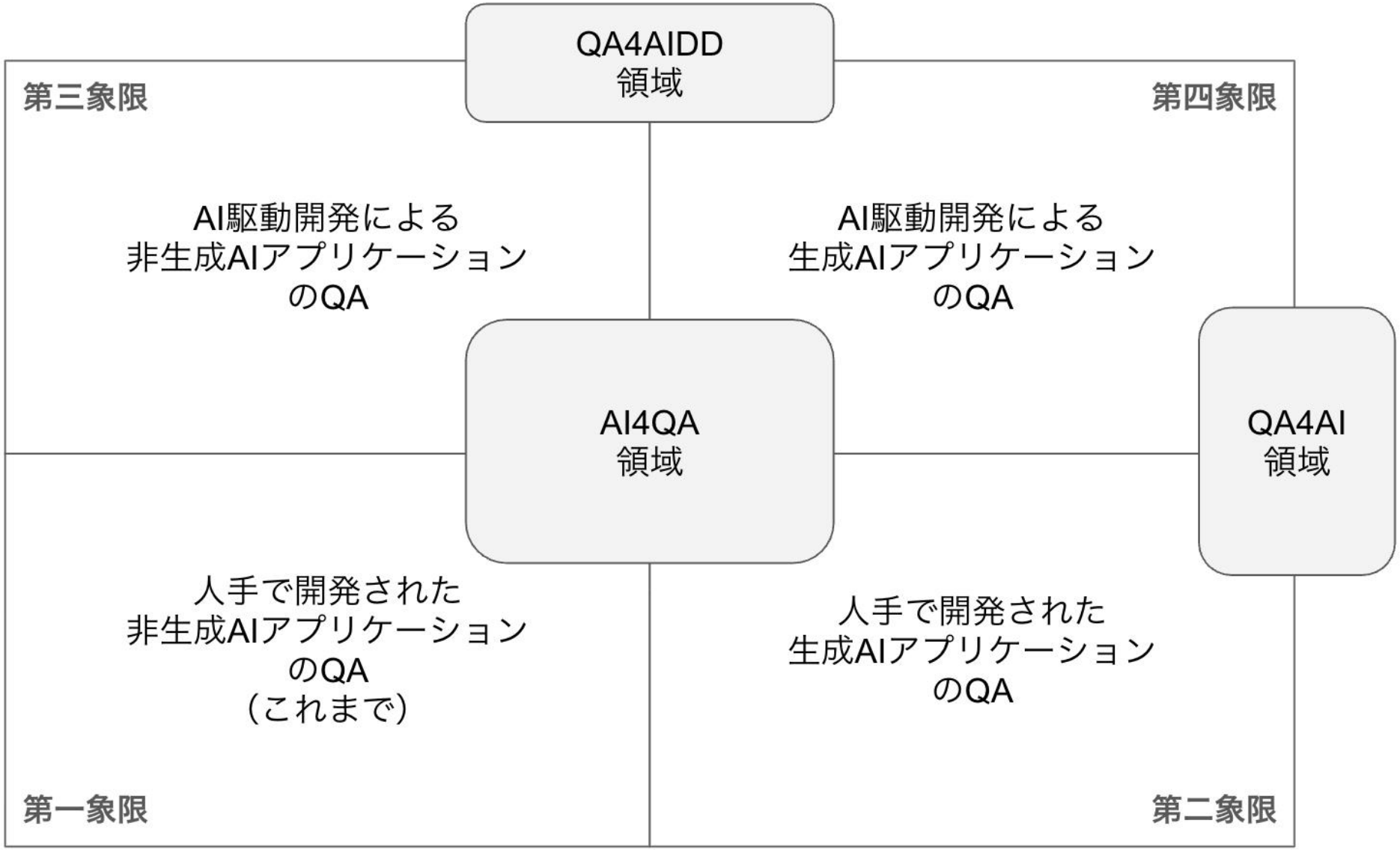
- コンパイラワーニングへの対応
- セキュアコーディングの遵守
- 静的解析（既知の脆弱性）
- ファジング（未知の脆弱性）

内部品質
(保守性、変更性)

- コーディングガイドラインの遵守
- コードレビュー指標の遵守
- コードメトリクスの計測

これらはプロセスQAによる担保が必要

QA x AI の関連モデル



出典：
「生成AIアプリケーションのテスト」（仮）