UWI Universal Wallet Infrastructure

# Agentic Trust Layer: AI エージェント時代のトラスト基盤を構築する

## Verifiable Credential とエージェント型 AI によるソフトウェアサプライチェーンガバナンス

共同著者:

NTT docomo Global　accenture　aws

# エグゼクティブサマリー

## 経営課題の本質

AI エージェントが契約を交わし、コードを書き、意思決定を行う時代において、「その AI は、誰が・どんな権限で動かしたのか」を改ざん不可能な記録として残し、規制当局・監査人・取引先が独立して検証できるインフラが、現時点では存在しません。これはセキュリティの問題ではありません。AI を事業の中核に置くためのトラスト基盤（Agentic Trust Layer）の欠如という、経営レベルの課題です。

既存の AI セキュリティ・ガバナンスソリューションの多くは、モデルの堅牢性とデータ入出力の制御に集中しています。しかし「AI が実際に何をしたか」をワークフローレベルで証明する第 3 層の仕組みは、業界全体で空白のままです。本ホワイトペーパーは、この空白に対する具体的な技術的回答を提供することを目的としています。

## 本取り組みが解決すること

本取り組みは、NTT DOCOMO GLOBAL とアクセンチュアが共同開発した UWI（Universal Wallet Infrastructure）の Verifiable Credential 技術と、AWS の AI プロダクト群を組み合わせることで、以下の 3 点を同時に実現するアーキテクチャを提案します。

**AI 生成コードの来歴証明：** どの AI が・誰の指示で・どのモデルを使って生成したコードかを、事後に誰でも独立して確認できる形で記録します。これにより、エンタープライズにおける AI 駆動開発の最大の阻害要因であった「コードの出自を追跡できない」という課題を解消します。

**継続的なセキュリティ保証の自動化：** コード変更のたびに脆弱性分析と信頼検証が自動実行されます。一度限りの監査ではなく、開発ライフサイクル全体にわたる継続的なガバナンスを実現します。

**将来のあらゆるエージェント活動への拡張：** AI 駆動開発を最初の実装領域としますが、同じ基盤は Agentic Commerce・採用 AI・医療・金融・製造業へそのまま適用で

きます。今この基盤を構築することが、AI 時代の信頼インフラにおける優位性につながります。

## 対象市場と規制動向

| 指標 | 数値 | 含意 |
|---|---|---|
| 国内ソフトウェアプロジェクト数 | 約 48 万 | 2027 年には約 7 万プロジェクトが、SBOM を含むソフトウェアサプライチェーン管理強化の対象となる可能性があります |
| 国内 SBOM 利用率 | 7% → 15%（2025→2027） | EU Cyber Resilience Act・経産省施策を背景に急拡大しています |
| グローバル検証トランザクション | 約 1.92 億件（2027 推計） | AI 駆動開発の普及でビルド頻度が加速し、市場規模は 2025 年比で倍増する見込みです |

EU Cyber Resilience Act（Regulation (EU) 2024/2847）では、2027 年 12 月に主要義務の適用開始が予定されており、ソフトウェアを含むデジタル要素を持つ製品に対するセキュリティ対応の重要性が高まります。SBOM はその最初の接点に過ぎません。取引先・規制当局が「証明できること」を前提とする時代において、対応の遅れは競争上のリスクに直結します。

## 本ホワイトペーパーの対象読者と活用方法

本ホワイトペーパーの主な対象読者は、AI 駆動開発の導入を検討または推進しているソフトウェアエンジニア、セキュリティエンジニア、アーキテクト、CISO、およびコンプライアンス担当者です。

経営層・CIO の方には、第 1 章および第 2 章をお読みいただくことで、Agentic Trust Layer という概念と AI 駆動開発における信頼課題の全体像を把握いただけます。技術的な詳細については、第 6 章（技術アーキテクチャ詳細）をご参照ください。

# 目次

# 1. Agentic Trust Layer — AI 時代のトラスト基盤

**なぜ「トラスト基盤」が経営課題になるのか**

AI エージェントが自律的に動く世界では、「そのエージェントは本当に権限を持っていたのか」「指示した人間は誰か」「何をどう判断したのか」という問いに答えられることが、企業活動の前提条件になります。規制当局はコンプライアンスの証拠を求め、取引先はサプライチェーンの透明性を求め、株主は AI ガバナンスの実態を問います。これらの問いに答えられない組織は、AI を事業の中核に置くための条件を満たしていないと見なされるリスクがあります。

**Agentic Trust Layer とは何か**

Agentic Trust Layer とは、AI エージェントのあらゆる活動に「誰が・どんな権限で・何をしたか」の証明を組み込む基盤インフラです。具体的には、エージェントの行動を改ざん不可能な記録として残し、その記録を規制当局・監査人・取引先が独立して検証できる仕組みを、開発・運用ワークフローに組み込むことで実現します。

既存の AI セキュリティ・ガバナンスソリューションの多くは、モデル自体の堅牢性（第 1 層）とデータ入出力の制御（第 2 層）に集中しています。Agentic Trust Layer が対象とするのは第 3 層、すなわち「AI が実際に何をしたかのワークフローレベルの信頼と説明責任」です。この第 3 層は、業界全体で現時点では空白となっています。

**最初の実装領域と拡張パス**

本取り組みは AI 駆動開発を最初の実装領域として選択しています。AI エージェントが最も活発に活動しており、かつガバナンス要件が明確に問われ始めている領域であるためです。しかし Agentic Trust Layer の設計原則は普遍的であり、同じ基盤がエージェントの活動するあらゆる領域に適用可能です。今この基盤を構築することが、AI 時代の信頼インフラにおける優位性につながります。

# 2. AI 駆動開発における信頼課題

前章では、Agentic Trust Layer というトラスト基盤の概念と、その最初の実装領域として AI 駆動開発を選択した背景を示しました。本章では、AI 駆動開発という領域において信頼課題がどのような形で顕在化しているかを具体的に整理します。

これらの信頼課題が特に顕著な領域がソフトウェア開発です。AI 駆動開発では、 AI を開発プロセスの中核的なコラボレーターとして位置づけ、計画の策定・実装・テストを AI が実行します。これにより、AI が生成するコードの来歴、エージェントの説明責任、ソフトウェア成果物の完全性という新たな問いが生じます。

従来の SBOM(Software Bill of Materials)はソフトウェア構成の可視性を提供しますが、AI 駆動開発の環境では新たな限界が露呈します。SBOM は「どのライブラリが含まれているか」は記録できても、「そのコードを誰が・どの AI が・どのような指示のもとで生成したか」は記録できず、SBOM ファイル自体の真正性を暗号学的に証明する仕組みも備わっていません。SBOM は広く採用されているものの、実際の開発・運用プロセスの中でリアルタイムに信頼を検証する仕組みとしてではなく、監査や規制対応のために事後的に作成される文書にとどまっており、その発行者の真正性の証明も運用上確立されていないのが現状です。 AI のセキュリティとガバナンスは 3 つの層で理解できます。第 1 層はモデルレベルのセキュリティ、第 2 層はデータの入出力に対するガバナンス、第 3 層はワークフローレベルの信頼と説明責任です。既存ソリューションの多くは第 1・第 2 層に焦点を当てていますが、本取り組みは第 3 層、すなわち運用ワークフローに検証可能な信頼と説明責任を組み込むことに焦点を当てています。

これらの課題に対処するため、本取り組みは分散型識別子(DID)と検証可能なクレデンシャル(VC)を信頼の基盤として採用します。このアプローチは、 SBOM の内容と発行者の真正性を暗号学的に検証できること、標準に準拠することで特定のベンダーやシステムに依存せず異なる組織間で検証できること、クレデンシャルの有効期限や無効化を管理できること、そして同じ仕組みを AI エージェントの行動の証明にも拡張できることを可能にします。既存の SBOM や SCA ツールを置き換えるのではなく、それらを強化する信頼レイヤーとして機能します。

## 3. 本取り組みの目標

自律型 AI エージェントがコードを生成・修正することが当たり前となった今、「誰が・どの AI が・どのような状況でコードを生成したか」を証明できないことが、エンタープライズにおける AI 採用の根本的な障壁となっています。本取り組みは、この課題に対して具体的な技術的回答を提供することを目的としています。

本取り組みは、UWI の Verifiable Credential (VC) および信頼技術と、AWS の AI プロダクト群を組み合わせることで、「AI トラスト」という新たな価値領域を創出する可能性を実証することを目指しています。AI 技術の急速な普及が進む中、AI 生成データや意思決定プロセスの信頼性をいかに確保するかが、重要な課題として浮上しています。

特に、エンタープライズソフトウェア開発において AI 支援・AI 主導によるコード生成(Agentic Coding)が主流となりつつある現在、AI 生成コードの出所、完全性、およびセキュリティが重大な懸念事項として認識されるようになっています。どの AI モデルが、誰の指示のもとで、どのような依存関係をもとにコードを生成したかを追跡できる検証可能な仕組みがなければ、組織はソフトウェアサプライチェーンのセキュリティおよびコンプライアンス対応において深刻なリスクにさらされることになります。

こうした背景のもと、本取り組みでは、UWI が提供する分散型アイデンティティ(DID)および検証可能なクレデンシャル技術を活用することで、AI エージェントのアイデンティティ検証、データの真正性保証、そしてエンドツーエンドのプロセス透明性を実現する新たなアプローチを検証します。

## 3.1 中核的な技術アプローチ: VC-SBOM

本取り組みの中核となる技術的アプローチは、ソフトウェア部品表(SBOM)を検証可能なクレデンシャル(VC-SBOM)として発行することです。ソフトウェアコンポーネント、依存関係、およびその出所を記録した SBOM を、UWI が管理する暗号署名済みの検証可能なクレデンシャルとして表現することで、人間が記述したコードと AI が生成したコードの双方について、改ざん検知可能かつ監査可能な記録を確立します。

さらに本取り組みでは、AI 生成コードに固有の情報を記録するために AI-SBOM という概念を導入します。AI-SBOM とは、通常の SBOM が記録するコンポーネントや依存関係の情報に加えて、「誰が AI に指示したか」「どの AI エージェントがコードを生成したか」「どのモデルバージョンが使用されたか」といった AI 固有のメタデータを追加で記録する拡張 SBOM です。

VC-SBOM と AI-SBOM は、それぞれ異なる軸で SBOM を拡張する概念です。VC-SBOM は信頼メカニズムの軸（SBOM を VC として署名・発行し、改ざん検知と発行者の真正性証明を付加する）、AI-SBOM は記録内容の軸（AI 固有のメタデータを追加で記録する）に対応します。この 2 つの軸は独立しており、本取り組みではその両方を組み合わせることで、AI 生成コードの来歴を暗号学的に検証可能な形で記録します。

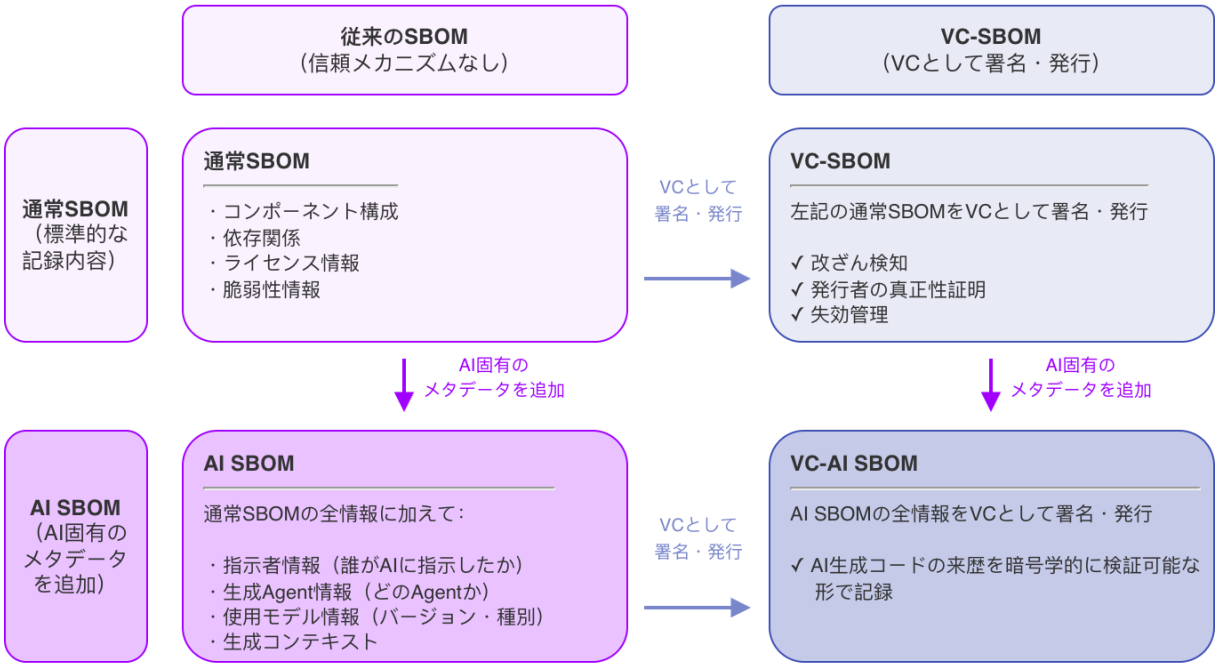


**図 1: VC-SBOM と AI-SBOM の関係**

このアプローチは自然に AI-SBOM へと拡張されます。AI-SBOM とは、通常の SBOM を AI 生成コードの追跡に特化して拡張したものであり、コードを生成した AI エージェントのアイデンティティ、使用したモデルのバージョンと種別、そして人間から AI への指示の連鎖(誰が・どのモデルで・どのような指示のもとで生成したか)を追加的に記録します。これにより、コードの発生源から承認・リリースに至るまでの各ステップが暗号学的に証明され、AI 生成コードに対する説明責任と監査可能性が確立されます。

AI-SBOM を AI 駆動開発全体に組み込むことで、開発ワークフローの重要なタイミング(コード生成・差分更新・リリース承認)において、セキュリティ検証と信頼証明が自動的に連動する信頼性の高い AI 開発・運用モデルを実現します。

## 3.2 AI エージェントへの Verifiable Identity 付与

SBOM によるコードの出所管理が確立されると、次の自然な発展として「そのコードを生成した AI エージェント自身のアイデンティティをどう証明するか」という課題が浮かび上がります。自律型 AI エージェントがコードの生成・レビュー・修正といったエンタープライズにおける開発ワークフローに積極的に関与するようになり、「どのエージェントが行動したか」を暗号学的に証明できる仕組みが、AI 駆動開発におけるガバナンスの前提となっています。

この課題へのアプローチとして、UWI を通じて AI エージェントに DID ベースの検証可能なアイデンティティ(Verifiable Identity)を付与することが有効です。これにより、マルチエージェント環境における委任チェーン(Planner Agent → Coder Agent → Reviewer Agent → Human Approver)を検証可能な記録として保存し、各エージェントの行動を組織のガバナンスポリシーに沿って監査できるようになります。 また、エージェントに Verifiable Identity が付与されていることで、異常な動作が検出されたエージェントを特定し、そのエージェントが発行に関与した VC のステータスを更新するといった対応も可能となります。

## 3.3 実証する価値提案

本取り組みでは、Verifiable Credential と AWS の AI サービスを組み合わせることで、以下の価値提案を具体的に実証します。具体的には、AWS が提供する仕様駆動開発を支援する統合開発環境である Kiro と、エージェント実行基盤である Amazon Bedrock AgentCore を活用します。

**SBOM の信頼性向上:** SBOM を Verifiable Credential として発行(VC-SBOM)することで、SBOM の内容が改ざんされていないこと、および発行者が真正であることを暗号学的に検証可能にします。これにより、SBOM は事後的に作成されるコンプライアンス文書から、開発・運用プロセスの中でリアルタイムに信頼を検証できる運用上のメカニズムへと転換されます。本取り組みでは、Analyzer Agent による VC-SBOM の自動生成と UWI による署名・検証の一連のプロセスを実装し、この転換が技術的に実現可能であることを示します。

**AI 生成コードの出所証明:** 誰が・どの AI モデルが・どの仕様に基づいてコードを生成したかを、改ざん不可能な VC-SBOM として記録・検証できること。本取り組みでは、AI-SBOM の拡張フィールド（指示者情報、AI エージェント情報、使用モデル情報、生成コンテキスト）を定義し、これらが VC-SBOM として正しく記録・検証されることを実装を通じて確認します。

**継続的なセキュリティ保証:** 脆弱性検知と SBOM 管理を AI 駆動開発のワークフローに統合し、一度限りの検証ではなく継続的なトラスト確保が実現できること。本取り組みでは、コードが生成・修正されるたびに SBOM 生成から脆弱性分析までが自動的に実行される一連のフローを実装し、継続的な保証が運用上成立することを確認します。

本取り組みで確立するトラストチェーンは、将来的には AI エージェント自身への Verifiable Identity 付与(DID によるエージェントの識別と、エージェント間の委任関係の記録)へと発展させることを想定しており、本取り組みはその第一ステップとして位置づけられます。

# 4. ユースケース全体像

セクション 3.1 で述べた技術アプローチに基づき、本取り組みでは以下の 4 つのユースケースを想定しています。UC1 を中核とし、UC2〜UC4 がそれぞれ異なる観点から信頼メカニズムを補完する構成です。

なお、本ホワイトペーパーでは表記を以下のとおり統一します。VC-SBOM は、通常の SBOM または AI-SBOM を VC として署名・発行したものの総称です。AI-SBOM は、通常の SBOM に AI 固有のメタデータを追加して拡張したものです。両者の関係はセクション 3.1 の概念図に示したとおり、直交する 2 つの軸に対応します。

## 4.1 UC1: VC-SBOM および AI-SBOM の自動生成・署名

本 UC は、ソフトウェアのコード変更に対して SBOM を VC 形式で自動的に発行・証明する仕組みを提供します。UC1 は以下の 2 つのサブユースケースで構成されます。

- **UC1a: VC-SBOM --- 通常 SBOM の VC 化**
  コード生成、コード変更、依存関係更新、プルリクエスト作成、リリース承認な

どの特定イベントをトリガーとして、そのコードの部品表（SBOM）が自動的に生成・更新され、UWI が発行する VC として署名されます。これにより、SBOM は単なるリストファイルではなく、改ざん検知可能な証明書付きの部品表となります。人間が記述したコードと AI が生成したコードの双方が対象であり、すべてのコード変更に対して統一的な信頼メカニズムが適用されます。

- **UC1b: AI-SBOM --- AI 固有メタデータの記録**
  AI 駆動開発においては、各 AI コード生成時に、使用モデル、関連ツール、プロンプト情報などの AI 固有のメタデータが個別に記録されます。これらの情報は、コードがリポジトリにプッシュされるタイミングで AI-SBOM として統合され、UC1a と同様に VC として署名・発行されます。セクション 3.1 で述べたとおり、AI-SBOM は通常の SBOM に AI への指示者、コードを生成した AI エージェント、使用モデルの情報を追加で記録することで、AI 生成コードのサプライチェーン全体にわたるトレーサビリティを確立します。
  本 UC は、後続の UC2〜UC4 すべての基盤となる中核ユースケースです。

## 4.2 UC2: コード安全性分析

UC1 で発行された VC-SBOM を分析対象のデータソースとして活用し、含まれるコンポーネントを脆弱性データベース(NVD、OSV など)と自動照合することで、AI 生成コードのセキュリティリスクを可視化・分析します。

AI 生成コードについては、AI-SBOM に記録された AI 固有の情報を追加の分析対象として活用できます。具体的には、使用モデル情報、エージェントを構成するソフトウェアコンポーネント、エージェントがアクセスしたツールや MCP サーバー、およびプロンプト情報を分析対象とし、脆弱性の有無や信頼性の低い外部コンポーネントへの依存がないかを検出します。さらに、これらの分析結果をエージェントの振る舞いへのプロアクティブなフィードバックとして活用し、例えば脆弱なライブラリをコード生成時に選択肢から除外するようエージェントの動作を改善することも可能となります。

## 4.3 UC3: コードの継続的なセキュリティ検証とチェック

開発ライフサイクル全体を通じてコードのセキュリティを継続的・自動的に監視します。SBOM を VC として管理することで、特定時点のコード状態を暗号学的に証明し

ながら継続的なセキュリティ検証を実現します。CI/CD パイプラインとの統合により、新たな CVE が公開された際の影響範囲の特定や、セキュリティポリシーに違反するデプロイメントの自動ブロックが可能となります。

AI-SBOM との統合という観点では、CI/CD パイプラインのチェック時に、標準的な脆弱性検証に加えて、「承認された AI エージェントによってコードが生成されたか」「承認されたモデルバージョンが使用されたか」「人間によるレビューが実施されたか」といった AI ガバナンス指向の検証も自動的に実施できます。これにより、コードの品質とセキュリティを継続的に保証しながら、AI 生成コードに固有のガバナンス要件にも対応することが可能となります。なお、本取り組みでは開発ライフサイクルの重要なイベントをトリガーとした監視を想定していますが、エージェントの実行時動作の監視(ランタイムモニタリング)は、将来的な拡張が必要な課題の一つにあげられています。

## 4.4 UC4: AI エージェントへの Verifiable Identity を活用した信頼性検証

AI-SBOM には、どの AI エージェントがコードを生成したかが記録されます。 しかし、この記録が意味を持つためには、「AI エージェントの ID が真正であること」を証明できなければなりません。

セクション 3.2 で述べたとおり、 UWI を通じて AI エージェントに DID（分散型識別子）を発行し、その DID に紐づく VC を付与することで、検証可能なアイデンティティ（Verifiable Identity）が確立されます。DID はエージェントを一意に識別するための技術的な仕組みであり、Verifiable Identity は DID と VC の組み合わせによって実現される「第三者が検証可能なアイデンティティ」という概念です。AI エージェントへの DID 発行は、UWI 基盤の将来的な拡張として想定されており、本取り組みのスコープに含まれます。

UWI を通じて AI エージェントの DID を AI-SBOM の「AI エージェント情報」フィールドに記録することで、AI-SBOM 内の AI エージェント情報は「自己申告」ではなく、UWI によって証明された検証可能な情報となります。これにより、不正または異常な動作を示す AI エージェントの特定と即時対応、規制当局や外部監査人への「AI 生成コ

ードに対する責任の所在」の証明、そして、エージェント間の信頼チェーンの可視化と管理が実現可能になります。

これらのユースケースを実現するアーキテクチャの全体像は以下のとおりです。

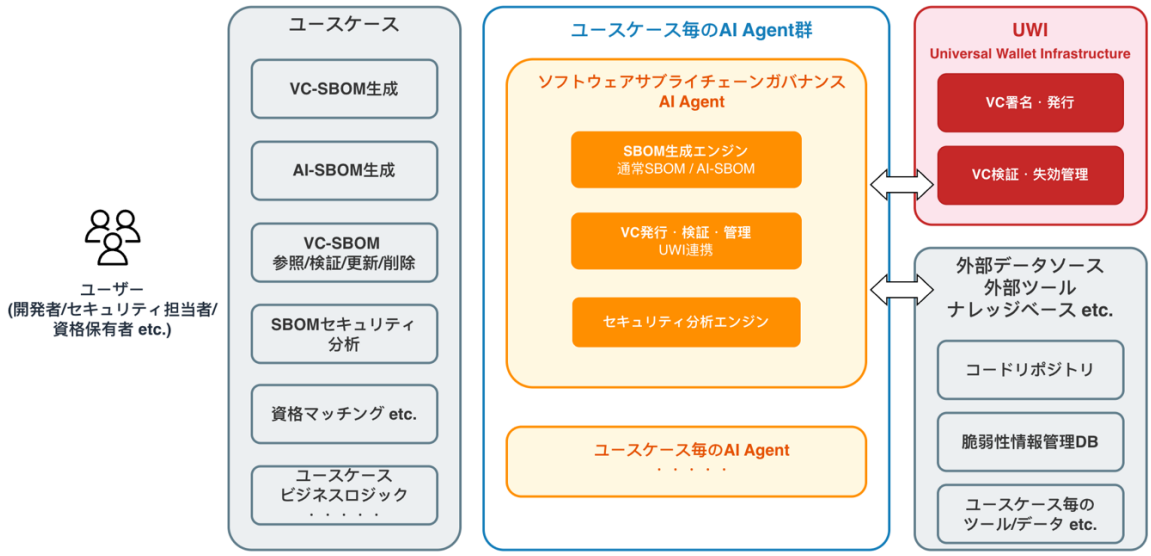


**図 2. ユースケースを実現するコンポーネント群**

# 5. 本取り組みのインパクトとビジネス視点

## 5.1 VC-SBOM の対象市場規模

本取り組みが対象とする市場の規模感を把握するために、SBOM はソフトウェアのビルド単位で生成・更新・検証されるため、ビルド回数の増加はそのままトランザクション規模の拡大を意味します。AI 駆動開発の普及によりビルド頻度が加速する中、この市場には相当な成長余地があると考えられます。

ソフトウェアプロジェクト総数は、グローバルの IT 技術者人口（約 3,000 万人、ヒューマンリソシア, 2024）を 3 人チーム換算し、グローバルで約 1,000 万、日本国内では技術者人口比から約 48 万と推計しています。SBOM 利用率は、国内では 2025 年時点で約 7%（ベリサーブ調査, 2025）、経済産業省の SBOM 活用促進施策や EU Cyber Resilience Act（CRA、Regulation (EU) 2024/2847）における 2027 年 12 月からの主要義務の適用開始予定を背景に 2027 年には国内 15%、グローバル 30%以上への拡大

が見込まれます（PwC/Black Duck）。ビルド頻度は、保守運用（月 1 回）と AI 駆動型高速開発（週 1 回）を合算した年 64 回を想定しています。

これらの前提をもとに計算すると、国内市場では 2025 年の約 215 万から 2027 年には約 461 万トランザクションへ、グローバル市場では 2025 年の約 9,600 万から 2027 年には約 1 億 9,200 万トランザクション程度の規模が想定されます。なお、上記は SBOM 生成・検証トランザクション全体の最大規模であり、このうち VC-SBOM として発行・検証される割合は VC-SBOM 基盤の普及度に依存します。

## 5.2 ステークホルダー別のインパクト

本取り組みは、ソフトウェアサプライチェーンに関わる 2 つのステークホルダー群に対して異なる価値を創出します。利用側である「システムの開発運用を行う組織および担当者」と、提供側である「ソリューションの実装と提供を行う組織」です。それぞれへのインパクトは、既存の通常 SBOM を対象とする領域と、AI 駆動開発とともに新たに生まれる AI-SBOM を対象とする領域の 2 軸で整理できます。

- **システムの開発運用を行う組織および担当者へのインパクト**
  通常 SBOM の領域では、人間が記述したコードと AI が生成したコードの双方について、SBOM を改ざん検知可能かつ監査可能な記録として確立することで、ソフトウェアの保守運用、インシデント対応、規制遵守が改善されます。これにより利用企業の競争力向上が期待されるとともに、信頼性の高い SBOM 運用が実現できるという確信が広がることで、SBOM そのものの普及もさらに拡大するものと想定されます。 AI-SBOM の領域では、AI 生成コードの来歴証明が可能となることで、コード生成における AI 活用に対するセキュリティやトレーサビリティの懸念が軽減されます。現在、エンタープライズにおける AI 駆動開発の採用における最大の阻害要因の一つが「AI が書いたコードの出自を追跡できない」ことへの懸念であり、AI-SBOM はこの懸念に対する直接的な解を提供することで、AI 駆動開発そのものの採用促進に貢献します。
  これらの価値が広く認知されるにつれて、VC-SBOM の発行・検証トランザクションの需要が拡大し、5.1 での想定される市場規模の実現を後押しすることが期待されます。

- **ソリューションの実装と提供を行う組織へのインパクト**
  通常 SBOM の領域では、UWI インフラ上で SBOM の生成・検証プロセスが提供されることで、SBOM ソリューション提供者の価値軸が拡張されます。従来、SBOM 情報を「記録できる」ことまでが十分な価値提供であったところから、暗号学的な「検証可能性」という別軸の付加価値への追従検討が想定されます。これは既存プレイヤーにとっては機能拡張の機会であり、同時に新規参入者にとっては差別化の起点となります。
  AI-SBOM の領域では、既存の SBOM 機能提供者に加えて、コーディングエージェントを提供する組織が AI-SBOM 機能の提供を検討することが想定されます。コーディングエージェント自身が生成プロセスの当事者であるため、来歴情報を最も自然に記録できる位置にいるためです。第三者による検証可能性がどの程度市場に求められるかは今後の動向に依存しますが、本取り組みは、暗号学的な検証可能性を備えた AI-SBOM を標準的な要件として確立することを目指しています。
  UWI にとっては、これらのソリューション提供者が VC-SBOM の発行・検証機能を自社サービスに組み込むことで、UWI インフラのトランザクション需要がエコシステム全体に拡大する構造が生まれます。

## 5.3 本取り組みにおける AWS の役割

AWS の AI 基盤を活用したソリューション実装は、5.2 で述べたステークホルダーへのインパクトを技術的に裏付け、さらに拡大します。

- **AI エージェント実装基盤としての価値**
  本取り組みのソリューションは、AWS が提供する AI エージェント向けの統合基盤上に構築されます。この基盤は、エージェントの開発・デプロイ・管理を一貫して行える環境を提供しており、ソリューションの実装者は信頼メカニズムの中核ロジックの開発に集中できます。インフラの構築・運用はマネージドサービスとして提供されるため、運用管理の工数を最小限に抑えながら、需要に応じたスケーラビリティを確保できます。また、AI 駆動開発のワークフローにエージェントを組み込みやすい設計となっているため、開発者が SBOM を意識的に管理する必要のない、シームレスな体験の実現を支えます。プラットフォーム自体が

セキュリティおよび信頼性の基本要件を満たしているため、利用者はソリューションの安全性に対する追加的な懸念なく導入を進めることができます。

- **VC-SBOM ワークロードに適した基盤としての価値**
  5.1 で試算したとおり、VC-SBOM の生成・検証トランザクションは年間数千万件から数億件規模に達する可能性があります。AWS のクラウド基盤は、このような大量トランザクションに対してスケーラブルかつ低レイテンシーな処理能力を提供するとともに、署名済み VC-SBOM を高い耐久性と改ざん耐性を備えたストレージに格納します。これにより、利用者は VC-SBOM の可用性と完全性の両面において安心してソリューションを利用できます。さらに、エージェントの動作ログや VC 発行履歴のトレース・監視を一元的に管理できるため、規制当局や外部監査人への説明責任を果たすための基盤としても機能します。技術的な詳細はセクション 6 で述べます。

# 6. 技術アーキテクチャ詳細

## 6.1 アーキテクチャの全体像

本セクションでは、セクション 4 で定義したユースケースのうち、UC1（VC-SBOM および AI-SBOM の自動生成・署名）、UC2（コード安全性分析）、UC3（継続的なセキュリティ検証）をカバーする技術アーキテクチャを説明します。UC4（Verifiable Identity を活用した信頼性検証）に関連する機能は、6.5（将来フェーズで対応する機能）で扱います。

本取り組みのアーキテクチャは、コンポーネントを機能的な役割に基づいて 2 つの層で整理しています。

**信頼関連コンポーネント(何を実現するか):** 本取り組みが設計・実装するコンポーネント群です。以下の 3 つの機能で構成されます。Analyzer Agent は、コード変更を契機にセキュリティ分析を実行し、分析結果を構造化された形で出力します。SBOM 生成エンジンは、Analyzer Agent の出力をもとに、VC-SBOM（通常 SBOM の VC 化）および AI-SBOM（AI 固有メタデータの記録）を生成します。UWI 統合インターフェースは、生成された SBOM を UWI に送信し、VC として署名・発行・検証する機能を提供

します。これらのコンポーネントが、VC ベースの SBOM 変換と AI 生成コードのトレーサビリティ記録という本取り組み固有のロジックを担います。

**実行環境と基盤サービス（どこで・どのように動作するか）:** 信頼関連コンポーネントが動作するための環境と基盤サービスです。開発環境としての Kiro、スケーラブルなエージェント実行基盤、署名鍵管理、アクセス制御などのクラウド基盤サービスで構成されます。本取り組みはこれらを組み合わせて活用し、信頼関連コンポーネントの安全かつ効率的な動作を実現します。各コンポーネントの詳細は 6.2 および 6.3 で説明します。

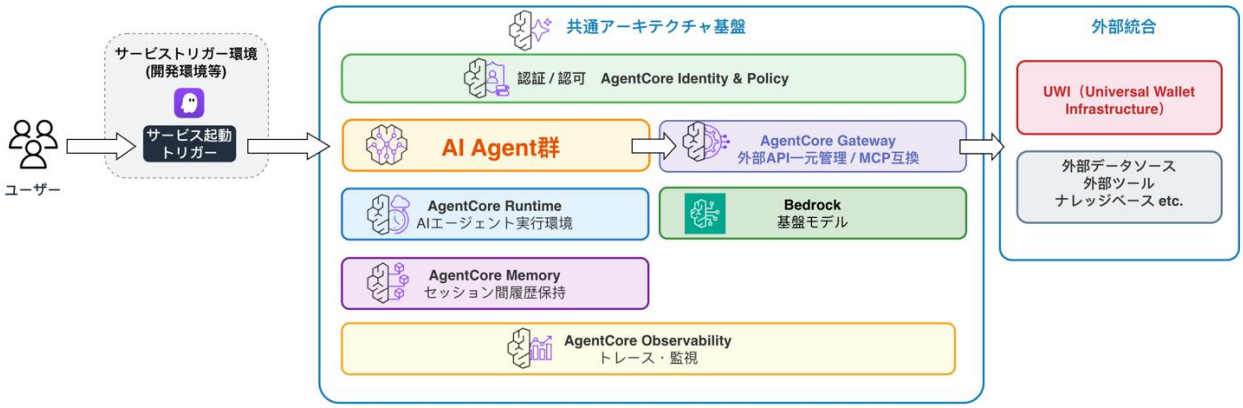


**図 3: 信頼コンポーネントが動作する環境と基盤サービス**

## 6.2 信頼関連コンポーネント

### 6.2.1 SBOM 生成エンジン

SBOM 生成エンジンは、コード変更を契機として、対象コードベースのコンポーネント情報を収集・構造化し、標準形式（CycloneDX または SPDX）の SBOM アーティファクトを生成するコンポーネントです。生成された SBOM は、6.2.2 で述べる UWI 連携を通じて VC として署名・発行され、VC-SBOM リポジトリに格納されます。本コンポーネントは UC1a（VC-SBOM）および UC1b（AI-SBOM）の実装を担います。

SBOM 生成エンジンは、通常 SBOM と AI-SBOM の 2 つのトラックを並行して管理します。

- **通常 SBOM**
  通常 SBOM は、コードがリポジトリにプッシュされたタイミングで、その時点のコードベースに対して生成されます。記録対象は、コンポーネントインベントリ（ライブラリ名、バージョン、依存関係、ライセンス情報）です。プッシュごとに当該時点のコードベースに対応する SBOM が 1 対 1 で生成され、VC-SBOM リポジトリにバージョン管理された履歴として格納されます。
- **AI-SBOM**
  AI-SBOM は、通常 SBOM と関連付けて管理する拡張的なトラックとして、AI 駆動開発に固有のメタデータを記録します。各 AI コード生成時に、以下の AI 固有の情報が個別に記録され、コードがリポジトリにプッシュされるタイミングで AI-SBOM として統合・生成されます。
  AI-SBOM に記録されるメタデータは以下のとおりです。
  - **指示者情報**：AI に指示を与えたユーザーの識別子、指示日時
  - **AI エージェント情報**：コードを生成した AI エージェントの識別子とバージョン（UC4 で Verifiable Identity が確立された場合は、DID を使用）
  - **AI モデル情報**：モデル名、バージョン、プロバイダー
  - **生成コンテキスト**：使用したプロンプト、参照したファイルやリソースの識別子、アクセスしたツールや MCP サーバーの情報
  - **Human-in-the-Loop 情報**：人間によるレビュー・承認の有無、承認者の識別子（UC4 導入後は DID を使用）

なお、プロンプト情報の記録形式（全文記録またはハッシュ値による記録）については、利用目的に応じて選択されます。過去のプロンプトとの同一性確認が目的であればハッシュ値で十分ですが、プロンプト内容の事後的な確認が必要な場合は全文記録が求められます。この選択は、組織のセキュリティポリシーやデータ保持要件に基づいて決定されます。ただし、大規模な開発組織における実運用では、プロジェクトやチームごとに異なる記録ポリシーを適用する必要が生じるため、ポリシーをプロジェクト単位で設定・管理できる仕組みが求められます。

### 6.2.2 VC 発行・検証・管理（UWI 連携）

本コンポーネントは、SBOM 生成エンジンが生成した SBOM、またはその参照情報を UWI（Universal Wallet Infrastructure）に送信し、W3C VC Data Model 標準に準拠し

た Verifiable Credential（VC）として署名・発行するインターフェースです。また、発行された VC-SBOM および AI-SBOM の VC-SBOM リポジトリへの格納、参照、失効管理も本コンポーネントが担います。UWI は署名鍵の安全な管理、VC の署名・発行・検証を行います。

- **VC 発行プロセス**
  VC 発行プロセスは以下のステップで構成されます。
  - **ステップ 1**（初期セットアップ）では、開発組織・プロジェクトおよび人間の開発者・承認者が UWI に識別子を登録します。UWI は各エンティティに Verifiable Identity Credential を発行します。なお、AI エージェントへの DID 発行については、UC4（将来拡張）として 6.5 で扱います。
  - **ステップ 2**（SBOM 送信）では、SBOM 生成エンジンが生成した SBOM アーティファクト（CycloneDX/SPDX JSON）本体またはその保存先の参照を UWI に送信します。リクエストには、SBOM アーティファクトまたはその保存先の参照、対象アーティファクトのハッシュ、AI-SBOM の場合は AI 固有メタデータ、およびリクエスト主体の識別情報が含まれます。
  - **ステップ 3**（署名・VC 構築）では、UWI がリクエスト元を認証した後、SBOM に対して署名を行い、W3C VC Data Model に準拠した VC を構築します。
  - **ステップ 4**（格納・配布）では、発行された VC が SBOM 生成エンジンに返却され、VC-SBOM リポジトリにバージョン管理された形で格納されます。VC 識別子は対応する Git コミットまたはプルリクエストにアノテーションとして付与されます
- **VC 検証プロセス**
  認可された検証者が UWI の検証機能を呼び出すことで、署名の有効性確認、発行者 DID の解決と確認、失効ステータスの確認が実施されます。これにより、対象の SBOM が改ざんされていないこと、正当な発行者によって発行されたこと、および現在も有効であることが検証可能となります。検証者のアクセス手段としては、API 直接呼び出し、CI/CD パイプラインからの自動呼び出し、管理コンソールからの手動確認など、複数のインターフェースが想定されます。

- **VC 失効管理**
  異常が検出された VC-SBOM または AI-SBOM の VC を失効させる必要がある場合、UWI の失効管理機能を通じて当該 VC が失効状態に更新され、以降の検証リクエストは失効済みのステータスを返します。

### 6.2.3 セキュリティ分析エンジン

セキュリティ分析エンジンは、VC-SBOM リポジトリに格納された VC-SBOM および AI-SBOM を分析対象として、コードのセキュリティリスクを検出・可視化するコンポーネントです。本コンポーネントは UC2（コード安全性分析）および UC3（継続的なセキュリティ検証）の実装を担います。

UC2 の観点では、VC-SBOM に含まれるコンポーネント情報を脆弱性データベース（NVD、OSV など）と自動照合し、既知の脆弱性を持つコンポーネントの検出を行います。AI 生成コードについては、AI-SBOM に記録された AI 固有の情報を追加の分析対象として活用できます。具体的には、使用モデル情報、エージェントを構成するソフトウェアコンポーネント、エージェントがアクセスしたツールや MCP サーバーの情報を分析し、脆弱性の有無や信頼性の低い外部コンポーネントへの依存がないかを検出します。

UC3 の観点では、CI/CD パイプラインとの統合により、新たな CVE が公開された際の影響範囲の特定や、セキュリティポリシーに違反するデプロイメントの自動ブロックを実現します。AI-SBOM との統合により、「承認された AI エージェントによってコードが生成されたか」「承認されたモデルバージョンが使用されたか」「人間によるレビューが実施されたか」といった AI ガバナンス指向の検証も自動的に実施できます。

分析結果は、脆弱性の通知やダッシュボードでの可視化に活用されるほか、エージェントの振る舞いへのプロアクティブなフィードバックとしても活用できます。例えば、脆弱なライブラリをコード生成時に選択肢から除外するようエージェントの動作を改善することが可能です。

### 6.2.4 Analyzer Agent（オーケストレーション）

Analyzer Agent は、6.2.1〜6.2.3 の各コンポーネントを統合し、開発環境からの呼び出しに応じて一連のプロセスを実行するオーケストレーションエージェントです。Amazon Bedrock AgentCore Runtime 上で動作します。

本取り組みでは、リファレンス実装の開発環境として Kiro を使用します。Kiro において所定のトリガー条件が満たされると、Analyzer Agent が呼び出されます。Analyzer Agent は、セクション 6.2.1 で述べたとおり、コードがリポジトリにプッシュされるタイミングで SBOM 生成エンジンを呼び出し、通常 SBOM および AI-SBOM の生成を開始します。ただし、Analyzer Agent の呼び出しインターフェースは特定の開発環境に依存しない設計であり、技術要件を満たす他のコーディングエージェントや開発環境からも同様に呼び出し可能です。

Analyzer Agent が実行する処理の流れは以下のとおりです。開発環境からの呼び出しを受信すると、まず SBOM 生成エンジン（6.2.1）を呼び出し、対象コードベースの通常 SBOM および AI-SBOM を生成します。次に、VC 発行・検証・管理コンポーネント（6.2.2）を通じて、生成された SBOM を UWI に送信し VC として署名・発行します。発行された VC は VC-SBOM リポジトリに格納されます。さらに、セキュリティ分析エンジン（6.2.3）を呼び出し、生成された VC-SBOM および AI-SBOM に対するセキュリティ分析を実行します。分析結果は通知やダッシュボードに連携されます。

将来的には、Analyzer Agent を含む信頼関連コンポーネント群を、他のエージェントや外部システムから呼び出し可能な MCP ツールとして公開することも想定しています。これにより、SBOM 生成・VC 発行・セキュリティ分析の各機能を、より広い範囲から利用可能となります。

## 6.3 実行環境と基盤サービス

### 6.3.1 Amazon Bedrock AgentCore（エージェント実行基盤）

Amazon Bedrock AgentCore は、6.2 で述べた信頼関連コンポーネント群（Analyzer Agent、SBOM 生成エンジン、セキュリティ分析エンジン）が動作する AI エージェント実行基盤です。特定のエージェントフレームワークに依存しない実行環境を提供し、

本取り組みでは Strands Agents（https://strandsagents.com/ ）をエージェントフレームワークとして採用しています。

本取り組みのアーキテクチャでは、AgentCore の以下の機能を活用します。

**AgentCore Runtime** は、信頼関連コンポーネント群を含む AI エージェントのサーバーレス実行環境です。セッション分離により、複数の開発者・プロジェクトの SBOM 生成・分析を安全に並行処理できます。長時間実行のサポートにより大規模コードベースの分析に対応するほか、軽量チェック向けの低レイテンシー実行環境も提供します。
**AgentCore Memory** は、プロジェクトごとの SBOM 履歴、脆弱性検知パターン、過去の分析結果をセッションをまたいで保持します。これにより、過去の分析結果を参照した効率的な分析が可能となります。

**AgentCore Gateway** は、UWI API や脆弱性データベース(NVD/OSV) API、など、外部サービスへの アクセスを一元管理するゲートウェイです。ツール呼び出しの認証・認可・レート制限を統合的に管理し、外部 API へのセキュアなアクセスを実現します。また、既存の API を MCP 互換ツールに変換する機能も持ち、標準化されたインターフェースを通じた将来的な機能拡張を可能にします。

**AgentCore Observability** は、信頼関連コンポーネント群の動作ログや分析結果 をトレース・監視します。エージェントの動作を可視化し、異常検知、パフォーマンス最適化、監査コンプライアンスに活用します。

**Strands Agents** は、言語モデル、システムプロンプト、ツールセットの 3 要素を組み合わせたモデル駆動アプローチでエージェントを構築できるオープンソースのエージェントフレームワークです。AgentCore Runtime と組み合わせることで、プロトタイプから本番環境へのシームレスなデプロイが可能となります。本取り組みでは、Analyzer Agent（6.2.4）の実装フレームワークとして採用しています。

### 6.3.2 VC-SBOM/AI-SBOM リポジトリ

Amazon S3(バージョニング有効)を、VC-SBOM および AI-SBOM の VC を格納・管理する VC-SBOM リポジトリとして採用します。発行された VC はバージョン管理された形で本リポジトリに格納されます。バージョン管理による履歴の保持と、CI/CD パ

イプラインとの統合が容易である点から、ストレージ実装として S3 を採用しています。

## 6.4 サービス呼び出しフロー

本セクションでは、 AI 駆動開発においてコードが生成・修正された後、一連のプロセスがどのような順序で実行されるかを説明します。本取り組みではリファレンス実装の開発環境として Kiro を使用することを想定しています。

1. コーディングエージェントまたは開発者によってコードが生成・修正される
2. 開発環境において所定のトリガー条件が満たされ、Analyzer Agent（6.2.4）が呼び出される
3. Analyzer Agent が SBOM 生成エンジン（6.2.1）を呼び出し、6.2.1 で述べたとおりプッシュ時点のコードベースに対して通常 SBOM および AI-SBOM を生成する
4. SBOM 生成エンジンが CycloneDX/SPDX 形式の SBOM アーティファクトを生成する
5. VC 発行・検証・管理コンポーネント（6.2.2）を通じて、生成された SBOM を UWI に送信する
6. UWI が SBOM に対して署名を行い、署名済み VC が SBOM 生成エンジンに返却される
7. VC が VC-SBOM リポジトリ（6.3.2）に格納される
8. セキュリティ分析処理がトリガーされる
9. セキュリティ分析エンジン（6.2.3）が、生成された VC-SBOM および AI-SBOM に対する脆弱性分析を実行する

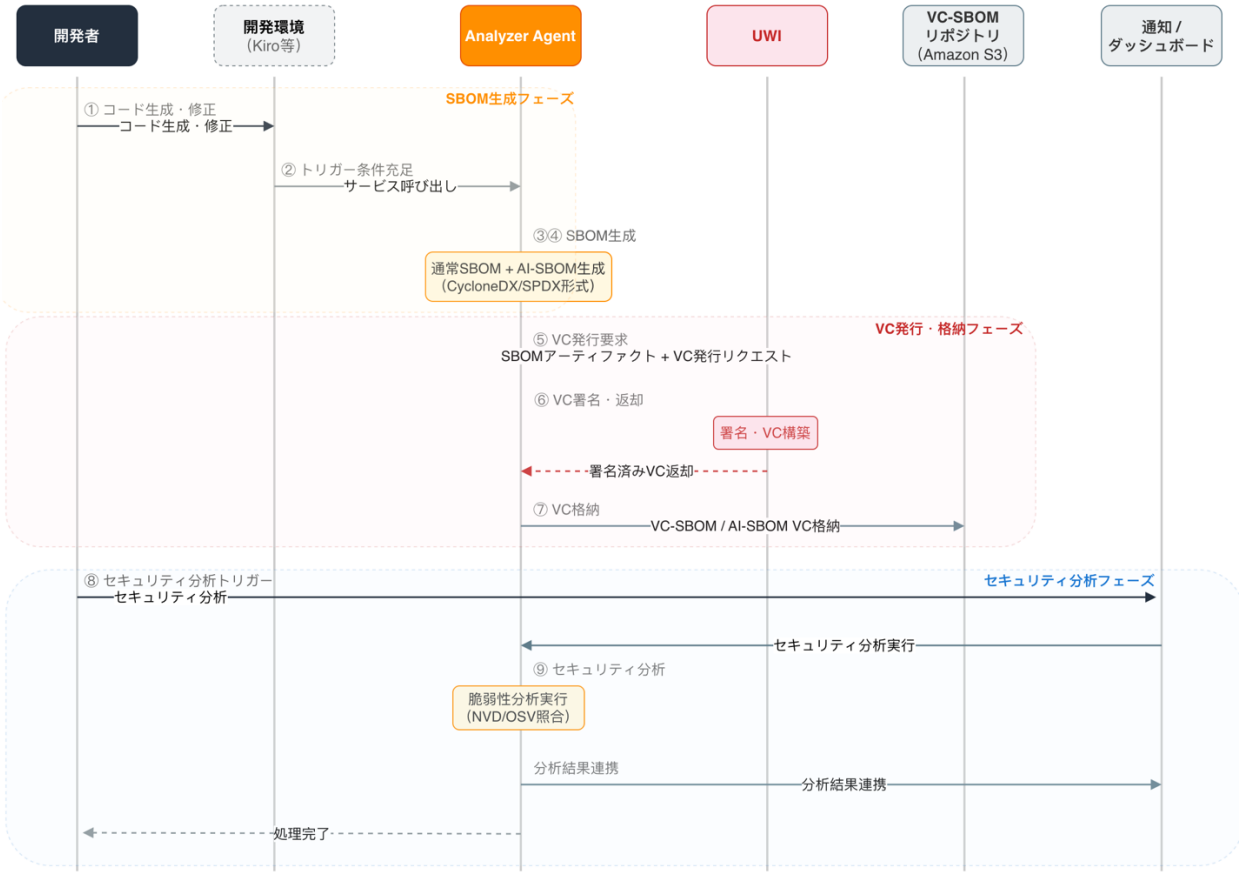


**図 4: サービス呼び出しシーケンス**

このフローにより、コードが生成・修正されるたびに、通常 SBOM に対しては VC-SBOM が、AI 生成コードに対しては AI-SBOM の VC が自動的に発行され、ソフトウェアサプライチェーン全体にわたるトレーサビリティが確立されます。

## 6.5 将来フェーズで対応する機能

本取り組みのアーキテクチャは、現在の実装範囲を超えて、以下の方向での機能拡張を想定しています。

- **UC4（AI エージェントの Verifiable Identity）の実装**
  セクション 4.4 で述べたとおり、AI エージェントへの DID 発行と Verifiable Identity の確立は、本取り組みの将来拡張として位置づけられています。UC4 が実装されることで、AI-SBOM の識別子フィールド（6.2.1）が DID に移行し、エージェント情報の第三者検証が可能となります。

- **CI/CD パイプラインとの統合**
  現在のアーキテクチャでは、開発環境からの呼び出しを契機とした SBOM 生成・VC 発行を対象としていますが、将来的には CI/CD パイプラインのイベント（プルリクエスト、マージ、リリース等）を契機とした自動化の拡張も想定されます。
- **信頼関連コンポーネント群の MCP ツール化**
  6.2.4 で述べたとおり、SBOM 生成、VC 発行・検証、セキュリティ分析といった本取り組みの信頼関連コンポーネント群が提供する一連の機能を、MCP ツールとして標準化し、他のエージェントや外部システムから呼び出し可能にすることを想定しています。これにより、本取り組みが構築した信頼メカニズムを、より広い範囲から利用可能となります。

# 7. 共通アーキテクチャへの拡張

## 7.1 拡張の考え方

セクション 6 で説明したアーキテクチャは、AI-SBOM によるソフトウェアサプライチェーンガバナンスという特定のユースケースに対して設計されたものです。本セクションでは、このアーキテクチャがどのように拡張可能であるかを、2 つの段階に分けて説明します。

第一の段階は、AI 駆動開発の領域内での拡張です。本取り組みが対象とする AI-SBOM は、AI 駆動開発における信頼メカニズムの一側面ですが、AI 駆動開発には SBOM 以外にも信頼の確立が求められる領域があります。例えば、UC4（セクション 4.4）で述べた AI エージェントへの Verifiable Identity 付与や、AI 生成コードのレビュー・承認プロセスの検証可能な記録などが挙げられます。セクション 6 で構築した信頼関連コンポーネント群（SBOM 生成エンジン、VC 発行・検証・管理、セキュリティ分析エンジン、Analyzer Agent）の設計パターンは、これらの領域にも適用可能です。具体的には、ドメイン固有のデータを収集・構造化するコンポーネントと、それを VC として発行・検証する UWI 連携コンポーネントの組み合わせという設計パターンは、AI 駆動開発における他の信頼メカニズムにも共通して適用できます。

第二の段階は、AI 駆動開発を超えた他のドメインへの拡張です。本取り組みのアーキテクチャの本質は、「構造化されたデータを Verifiable Credential として発行・検証する」という信頼メカニズムにあります。この仕組みは、信頼性の高いデータ記録と検証が求められるあらゆるドメインに適用可能です。7.2 では、その具体例として Worker Credential（資格情報管理）ユースケースを紹介します。

## 7.2 拡張ユースケースの例: Worker Credential

AI 駆動開発を超えた拡張先の一つとして、Worker Credential（資格情報管理）ユースケースが挙げられます。

Worker Credential ユースケースでは、語学資格や専門資格といった資格情報を VC として発行・管理することに意味があります。資格発行機関（Issuer）が発行した資格情報を VC として記録することで、資格の真正性が暗号学的に検証可能となり、採用企業や教育機関（Verifier）は資格保有者（Holder）に対して直接確認を取ることなく、資格の有効性を独立して検証できます。また、Holder のプライバシーを保護しながら、必要な資格情報のみを選択的に開示するといった運用も可能となります。

本取り組みのアーキテクチャとの共通性の観点では、以下の設計パターンが再利用可能です。ドメイン固有のデータを収集・構造化するコンポーネント（AI-SBOM における SBOM 生成エンジンに相当）は、Worker Credential では資格情報の収集・構造化を担います。VC として発行・検証する UWI 連携コンポーネントは、W3C VC Data Model 標準に準拠した VC 発行・検証という基本的な信頼メカニズムとして、ユースケースを問わず共通して機能します。また、エージェント実行基盤（Amazon Bedrock AgentCore）上で動作するエージェントの実装方式も共通であり、Worker Credential における資格マッチングエージェント等も同じ実行基盤上に構築できます。

このように、本取り組みで確立したアーキテクチャは、ドメイン固有のビジネスロジック（AI-SBOM ではコード分析と SBOM 生成、Worker Credential では資格情報の収集とマッチング）を差し替えることで、異なるドメインのユースケースに拡張可能です。

## 7.3 拡張の設計原則

共通アーキテクチャを新しいユースケースへ拡張する際の設計原則を以下に示します。

**再利用性:** UWI 連携による VC 発行・検証パターン、エージェント実行基盤、MCP プロトコルによる標準化された連携方式は、新しいユースケースでも変更なく再利用できます。

**拡張性:** 新しいユースケースを追加する際は、ドメイン固有のデータ収集・構造化コンポーネントとそのデータに対する VC 発行ロジックを実装するだけで、既存の基盤に統合できます。

**相互運用性:** W3C DID/VC 標準と A2A/MCP プロトコルへの準拠により、異なるユースケースのエージェントが同じプロトコルで連携できます。 AI-SBOM の Analyzer Agent と Worker Credential の資格マッチングエージェントが情報を交換することで、「特定の資格を持つ開発者が AI エージェントを使用して生成したコード」の来歴を、資格情報とコード来歴の両面から検証するといった、ドメインを横断した信頼メカニズムの構築が将来的に可能となります。

**ガバナンスの一貫性:** すべてのユースケースにわたって、同一の UWI トラストアンカーを使用することで、 VC の発行者の真正性、データの改ざん検知、失効管理といった信頼メカニズムが統一された基準で運用されます。これにより、例えば AI-SBOM と Worker Credential のように異なるドメインの VC であっても、同一の検証プロセスで真正性を確認でき、組織全体で一貫したガバナンスフレームワークを維持できます。

# 8. 将来展望

## 8.1 本取り組みが示す方向性

本ホワイトペーパーで説明したアーキテクチャは、AI 駆動開発におけるソフトウェアサプライチェーンガバナンスという具体的な課題に対する技術的な回答を提供するものですが、同時により広範な問いへの入口でもあります。その問いとは、「AI が自律的に行動する世界において、信頼をどのように構造化するか」というものです。

本取り組みを通じて明らかになった重要な洞察は、信頼は事後的に付与するものではなく、ワークフローに組み込まれるべきものであるという点です。VC-SBOM と AI-SBOM は、コードが生成・修正・リリースされるたびに自動的に信頼の記録を生成し

ます。これは、監査のために後から証拠を収集するアプローチとは根本的に異なります。

## 8.2 拡張の方向性

本取り組みで確立した共通アーキテクチャは、複数の方向に拡張できます。

**ユースケースの拡張:** セクション 7.2 で説明した Worker Credential をはじめ、構造化された活動データと Verifiable Credential の組み合わせが価値を持つあらゆるドメインへの適用が考えられます。医療機器のソフトウェア認証、金融サービスにおける取引の来歴証明、製造業におけるサプライチェーンの透明性確保など、規制要件が厳しく監査可能性が求められる領域は特に有望な拡張先です。

**エージェントエコシステムの拡張:** UC4 で想定した AI エージェントへの Verifiable Identity 付与が普及するにつれて、エージェント間の信頼チェーンを検証可能な形で記録するインフラストラクチャとしての UWI の役割が拡大します。複数の組織にまたがるマルチエージェントワークフローにおいて、各エージェントの行動を暗号学的に証明できる仕組みは、エンタープライズ AI ガバナンスの基盤となります。

**データレイヤーの深化:** 本取り組みは構造化された活動データ（SBOM イベント、脆弱性検知イベント等）を中心としていますが、将来的には非構造化データ（設計ドキュメント、レビュー会議録画、会話ログ等）との統合により、より深い文脈理解に基づく信頼の確立が可能となります。構造化データのみでは「何をしたか」のパターン認識にとどまりますが、非構造化データを統合することで「なぜそうしたか」という文脈を理解した推論が実現されます。

## 8.3 標準化とエコシステムへの貢献

本取り組みでは、W3C DID/VC、CycloneDX/SPDX 形式の SBOM、A2A/MCP など、相互運用性を重視した標準・仕様・プロトコル群を活用します 。これにより、特定のベンダー実装に依存しすぎることなく、将来的なエコシステム連携や標準化動向への追随を容易にします。これらの標準への準拠は、特定のベンダーへの依存を避けながら、より広範なエコシステムとの相互運用性を確保するための重要な選択です。

また、本取り組みで確立するリファレンスアーキテクチャは、AI 駆動開発におけるソフトウェアサプライチェーンガバナンスの具体的な実装例として、業界標準の形成に貢献することを目指しています。特に、以下の 2 つの領域は現時点では標準化が十分に進んでいない領域であり、本取り組みの成果が将来の標準化議論に貢献できると考えています。

**AI-SBOM のデータモデルと VC 発行フロー:** AI 生成コードに固有のメタデータ(指示者情報、AI エージェント情報、モデル情報、委任チェーン等)を SBOM としてどのように表現し、Verifiable Credential として発行・検証するかについては、業界横断的な標準が確立されていません。本取り組みで定義する AI-SBOM のデータモデルと VC 発行フローは、この領域における具体的な実装例として標準化議論に貢献できます。

**AI エージェントへの DID 付与:** 自律型 AI エージェントに対して DID(分散型識別子)を付与し、エージェントの行動を暗号学的に証明可能にする仕組みは、現時点では標準化が十分に進んでいない領域です。マルチエージェント環境における委任チェーンの表現方法、エージェントのアイデンティティライフサイクル管理、失効メカニズムなど、解決すべき課題が多く残っています。本取り組みで実装する UC4 のアーキテクチャは、この領域における先行事例として、W3C DID 仕様や Verifiable Credentials Data Model の拡張議論に貢献できると考えています。

## 8.4 まとめ

AI が開発ワークフローの中核的なアクターとなった今、ソフトウェアサプライチェーンの信頼性を確保するための新たなアプローチが必要です。本ホワイトペーパーで提案する VC-SBOM と AI-SBOM のアーキテクチャは、この課題に対する具体的かつ実装可能な回答を提供します。

DID/VC という成熟した信頼技術と、Amazon Bedrock AgentCore というスケーラブルなエージェント実行基盤を組み合わせることで、AI 生成コードの来歴証明、継続的なセキュリティ保証、そしてエンタープライズガバナンスへの適合を同時に実現できます。

本取り組みは AI 駆動開発という特定の領域から始まりますが、その設計原則と共通アーキテクチャは、信頼性の高いデータ記録と検証が求められるあらゆるドメインへの拡張を念頭に置いています。AI が社会インフラの一部となる未来に向けて、検証可能な信頼の仕組みを今から構築することが、持続可能な AI 採用の基盤となると確信しています。

UWI Universal Wallet Infrastructure

docomo Global accenture aws

Learn more
universalwalletinfra.com

Get in touch
ndg_uwi@ml.nttdocomo.com