

Part 2

システム開発プロセスへの生成 AI 活用

日立製作所のシステム開発における 生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

生成 AI を活用することで、システム開発の効率を飛躍的に向上させたい。2022 年 11 月の ChatGPT 登場以降、このように考える企業は増えているのではないのでしょうか。実際に開発現場で生成 AI の活用を成功させている事例が登場している一方で、具体的にどのように活用すればいいのかわからない、生成 AI を提供してもその活用がなかなか浸透しない、という問題に直面しているケースも少なくないようです。

本ホワイトペーパーでは、日立製作所 (以下、日立) における GitHub Copilot の活用を中心に、システム開発現場での生成 AI 活用について、次の 3 つのパートに分けてご紹介します。

- Part 1 GitHub Copilot 導入の背景および社内検証
- Part 2 システム開発プロセスへの生成 AI 活用
～「日立アプリケーションフレームワーク Justware」との連携～
- Part 3 生成 AI のナレッジ共有と人財育成の取り組み
～「生成 AI 実務者コミュニティ」の活動を中心に～

CONTENTS

2-1	Justware OSS ベースとは.....	2
2-2	Justware OSS ベースにおける生成 AI 活用パターン 1: 業務ロジックのコーディングにおける GitHub Copilot の活用.....	5
2-3	Justware OSS ベースにおける生成 AI 活用パターン 2: Hitachi GenAI System Development Framework との連携.....	7
2-4	今後の展望.....	9

Part 2 システム開発プロセスへの生成 AI 活用

Part 1 でご紹介したのは「GitHub Copilot 単体での活用」の実証実験でしたが、開発業務における生産性向上効果をさらに高めていくには、これをシステム開発プロセス全体に組み込んでいく必要があります。システム開発を日常業務として遂行している企業や組織は、標準化された開発フレームワークや開発プロセスを持っていることが一般的であり、そのプロセスをどれだけ短縮し、品質を維持しながら省力化できるのかが重要になるからです。

日立製作所（以下、日立）でもこのような取り組みを行っています。そこで本 Part 2 では、大規模開発の効率化と品質確保を支援する「日立アプリケーションフレームワーク Justware OSS^{*1} ベース」（以下、Justware OSS ベース）に、生成 AI を掛け合わせた取り組みについてご紹介し、要件定義からシステムテストまでの開発プロセス全体に対する生成 AI の活用について深掘りします。

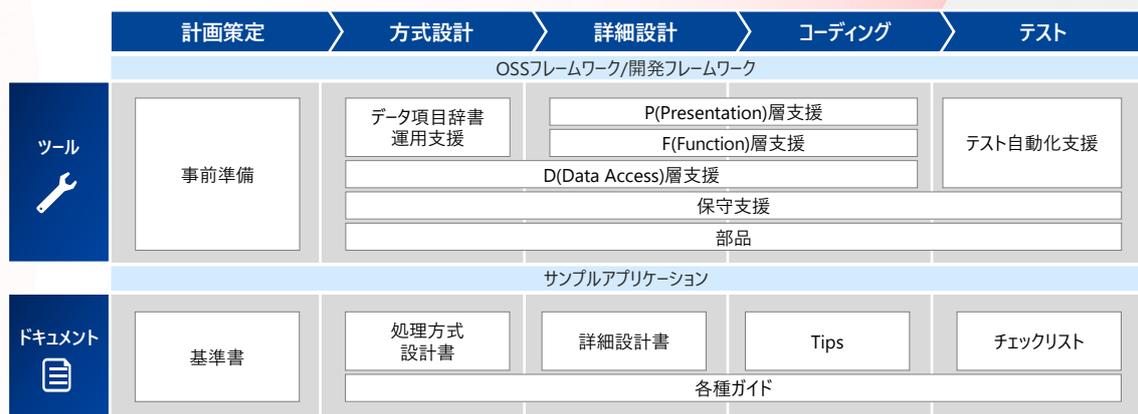
2-1 Justware OSS ベースとは

日立では多岐にわたるシステム開発を手掛けており、それらで利用している開発フレームワークも複数存在します。その中で、OSS を活用したアプリケーション開発の効率化と品質確保を支援する開発フレームワークが「Justware OSS ベース」です。その全体的な構成を示したのが下の図です。

日立アプリケーションフレームワーク Justware
Justware OSSベースとは

HITACHI

OSSを活用したアプリケーション開発の 効率化と品質確保を支援する開発フレームワーク



©Hitachi, Ltd. 2025. All rights reserved

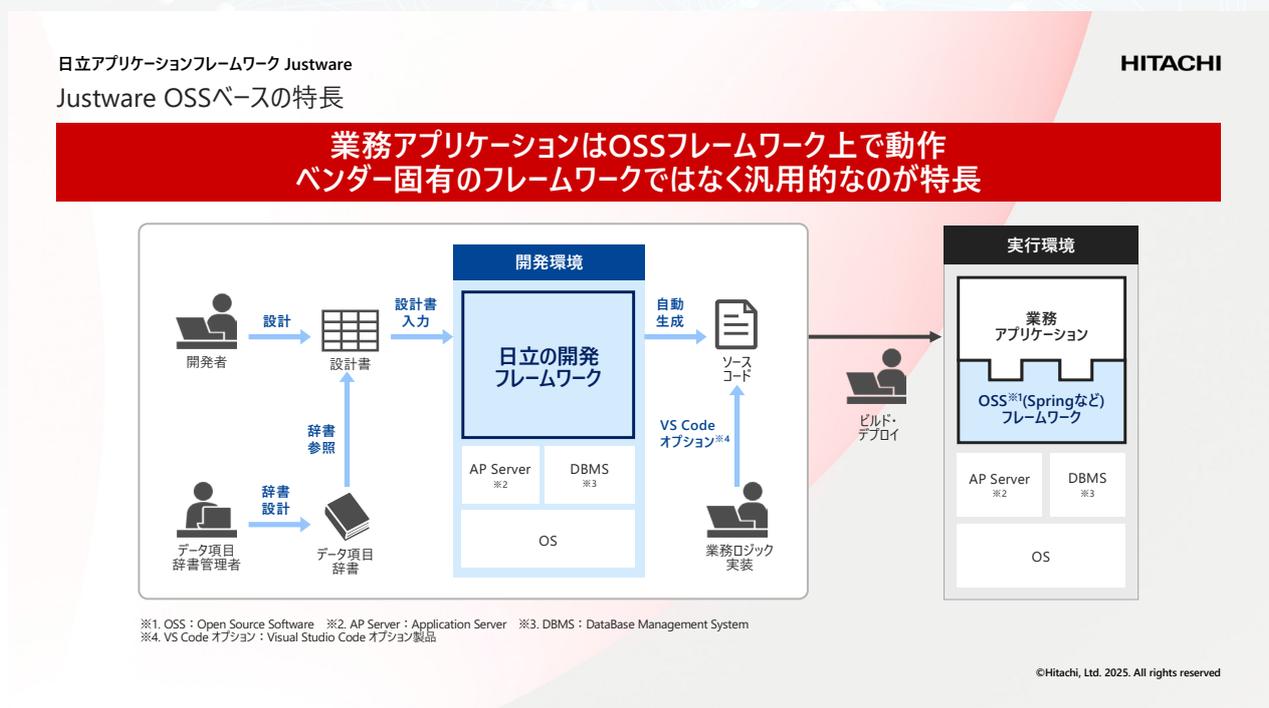
※1 OSS: Open Source Software。ソースコードが公開されており、再配布や拡張が可能で、主にコミュニティによる開発、拡張、更新が行われているソフトウェアのこと。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 2 システム開発プロセスへの生成 AI 活用

Justware OSS ベースは、計画策定から方式設計、詳細設計、コーディング、テストまでカバーしており、それぞれで利用するツール類や、作成すべきドキュメントが明確に定義されています。これによって作成されるアプリケーションは、「P (Presentation) 層」「F (Function) 層」「D (Data Access) 層」で構成されている、一般的な 3 層アーキテクチャのアプリケーションとなります。

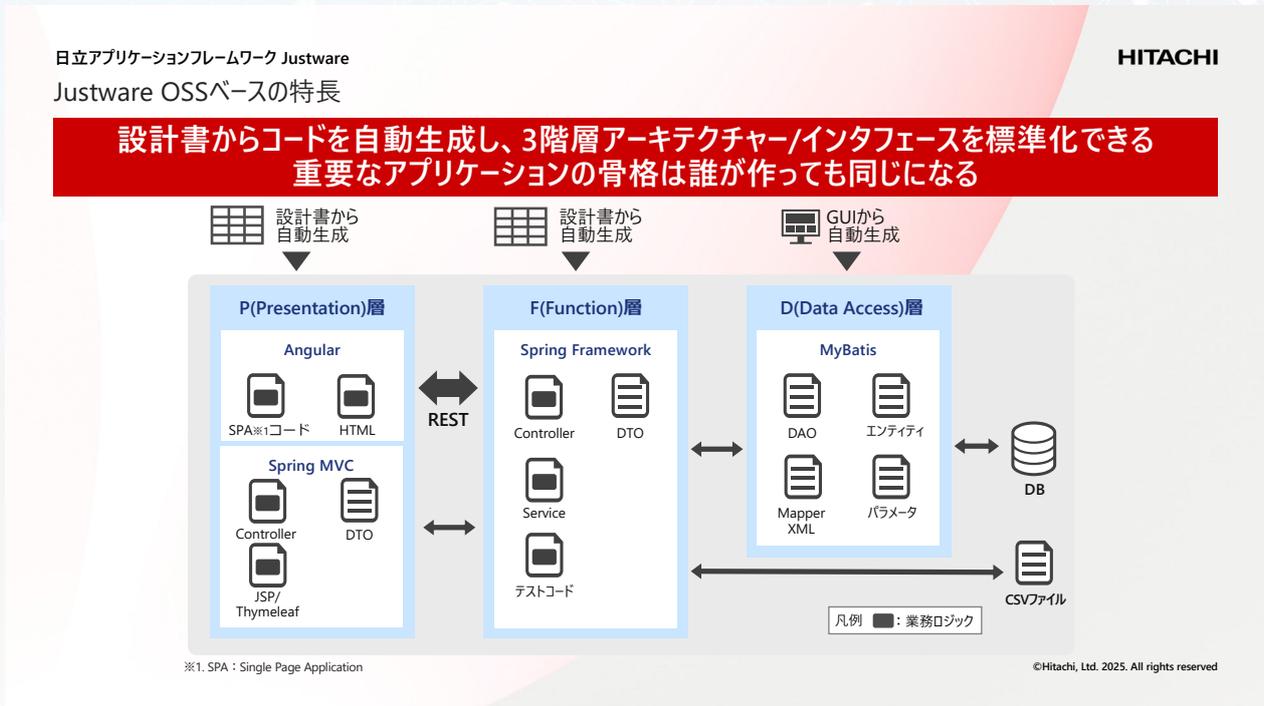
Justware OSS ベースで作成するアプリケーションは、「OSS フレームワーク」の利用を前提としています。そのためベンダー固有のフレームワークではなく、汎用的になっている点が大きな特長です。



Justware OSS ベースには、もう 1 つ重要な特長があります。それは、詳細設計書からルールベースで統制の取れたコードを自動生成する点です。自動生成されるのはアプリケーションの「骨格」部分であり、P 層の画面構成や F 層の業務ロジックは、人間 (プログラマー) が記述する必要があります。なお D 層のコードやパラメーターは、100% 自動生成されるようになっています。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 2 システム開発プロセスへの生成 AI 活用



こうした特長により、Justware OSS ベースは大規模なシステム開発においても、ルールベースで高精度なソースコードを自動生成できます。さらに、生成 AI の活用によって、開発の一層の高度化を目指す取り組みも進められています。例えば日立内では、システム開発プロセスへの生成 AI の活用方法として、以下の 2 パターンを実施、検証してきました。

1. 業務ロジックのコーディングにおける GitHub Copilot の活用
2. Hitachi GenAI System Development Framework との連携

ここからは、これら 2 つのパターンについて解説していきます。

2-2 Justware OSS ベースにおける 生成 AI 活用パターン 1: 業務ロジックのコーディングにおける GitHub Copilot の活用

前述のように、Justware OSS ベースでは詳細設計書からルールベースでアプリケーション骨格部分のコードを自動生成しますが、業務ロジックはプログラマーが作成しなければなりません。この業務ロジック作成の部分を効率化するために、GitHub Copilot を活用しているのがパターン 1 です。

Justware における生成 AI 活用パターン

業務ロジックのコーディングにおける GitHub Copilot の活用

HITACHI

パターン 1 : ルールベースの自動生成と組み合わせてコーディングする場合

- 手組みの業務ロジック部分は、エディター上でソースのコメントからコードを生成するなど、GitHub Copilot を活用してプログラマーを補佐。
- Justware OSS ベースの連携オプション(Visual Studio Code オプション)を利用することで業務ロジックのひな型生成。Microsoft Visual Studio Code 上でソースを自動生成し、詳細設計から GitHub Copilot の補助となるコメントを生成。



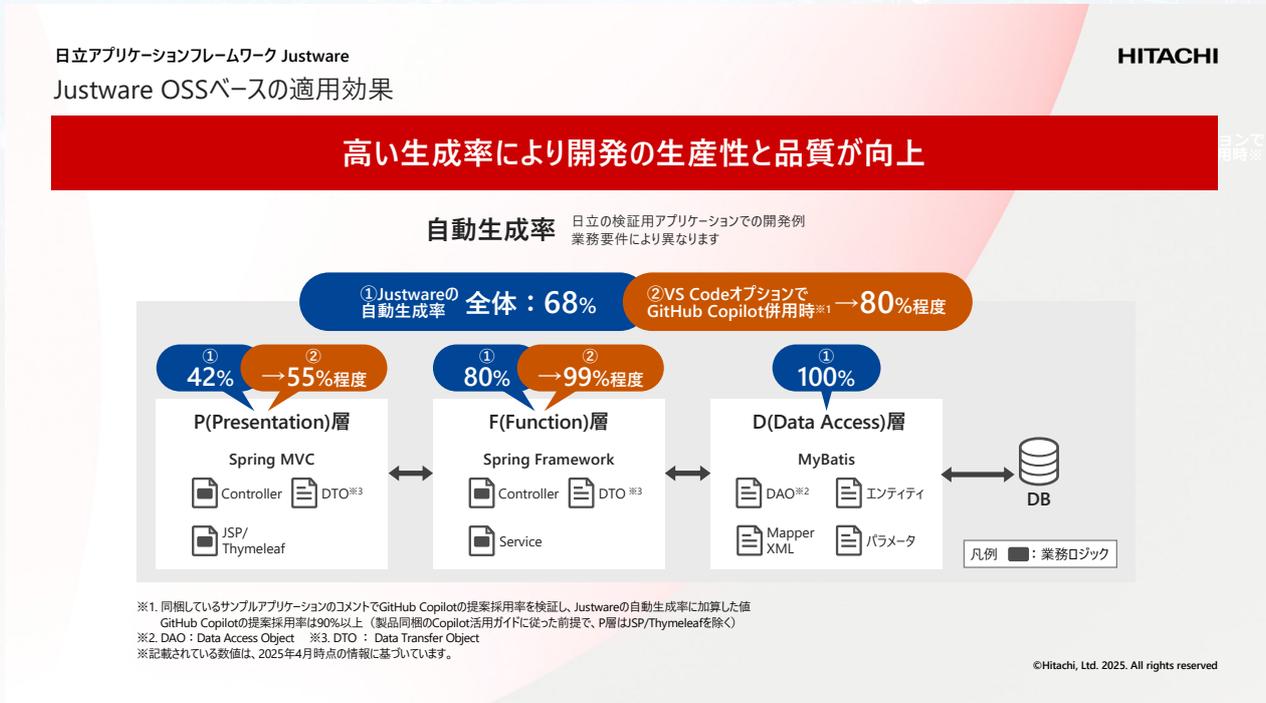
©Hitachi, Ltd. 2025. All rights reserved

具体的には、Justware OSS ベースに「Visual Studio Code オプション」を追加することで、プログラマーは Visual Studio Code 上で業務ロジックのコーディングを行えるようになるとともに、GitHub Copilot の利用も可能になります。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 2 システム開発プロセスへの生成 AI 活用

このようなプログラマー支援のための GitHub Copilot の効果を示す一例が以下の図です。



これは日立的検証用アプリケーションでの開発例になります。GitHub Copilot の支援は主に F 層の業務ロジック作成で活用されています。P 層はさきわめて自由度が高いためエンドユーザーを交えた試行錯誤が必要です。

F 層でのコード自動生成率は、GitHub Copilot を使用しない場合には 80% でした。これに対し、Justware OSS ベースで生成したコードを GitHub Copilot に読み込ませたうえで業務ロジックを生成させたところ、自動生成率が 99% にまで上昇しました。全体としては、68% だった自動生成率が 80% 程度にまで上昇したことになります。また自動生成率が高くなったことでヒューマンエラーを回避しやすくなり、アプリケーションのコード品質も向上しています。

もちろんこれは、このプロジェクト固有の数値であり、どの程度の効果が得られるかはプロジェクトの内容によって左右されます。骨格部分の比率が大きく、業務ロジックの割合が小さい場合には、生成 AI を利用しても自動生成率の上昇は限られてしまうでしょう。その一方で、骨格部分に比べて業務ロジックの割合が大きいプロジェクトでは、自動生成率は大きく向上することが期待できるでしょう。

Justware OSS ベースで GitHub Copilot を利用可能にする「Visual Studio Code オプション」は、2024 年 4 月から一般発売を行っており、適用が進められています。

2-3 Justware OSS ベースにおける 生成 AI 活用パターン 2： Hitachi GenAI System Development Framework との連携

パターン 1 と並行して、パターン 2 の検証も進められてきました。パターン 2 では、Hitachi GenAI System Development Framework と連携し、システム開発の全工程において生成 AI の活用を推進しています。具体的には、Justware OSS ベースで高品質なコードを自動生成するための準備作業や、テスト工程におけるテストケース・テストデータの作成などに生成 AI を活用する取り組みです。

The screenshot shows the Hitachi GenAI System Development Framework interface. On the left is a sidebar menu with categories like SE Assistant, Code Assistant, and QA Assistant. The main area is titled 'ソース生成' (Source Generation) and includes a 'プロンプト' (Prompt) section with a '生成対象' (Generation Target) dropdown set to 'Java(Spring Framework)'. A 'プロンプト入力支援メニュー' (Prompt Input Support Menu) is visible, along with a '生成結果' (Generation Result) section showing a code snippet for a 'Book' class. Annotations in Japanese highlight various features: '5つの目的メニュー' (5 target menus) pointing to the sidebar, '「SE」、「コード」、「PM」、「QA」、「共通(付帯業務)」をアシストする機能' (Function to assist with SE, Code, PM, QA, and Common tasks), '言語・フレームワークの選択' (Language/Framework selection), 'プロンプト入力支援メニュー' (Prompt input support menu), 'プロンプトの詳細説明および記載例を確認可能' (Can check detailed descriptions and examples of prompts), '生成AI回答表示フィールド' (AI answer display field), '個人別プロンプト管理(テンプレート・履歴)' (Individual prompt management), and 'Downloadボタンなど' (Download button, etc.).

Hitachi GenAI System Development Framework は、システム開発の全工程を対象に生成 AI を活用したさまざまなユースケースを提供しています。一方で、「生成 AI のハルシネーション (誤情報生成)」という課題もあり、特にミッションクリティカルシステムへの適用においては、既存の開発ツールとの連携は不可欠なものです。そこで、Justware OSS ベースの統制された高品質なルールベースツールや開発プロセスと連携し、生成 AI とルールベースの組み合わせによる精度と効率の両立に取り組んでいます。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 2 システム開発プロセスへの生成 AI 活用

例えば、基本設計で確定した仕様をもとに、Justware OSS ベースに適合した詳細設計書を作成する作業は、従来は人手で行う必要がありました。ここに生成 AI を活用することで、「高品質な自動生成を行う Justware OSS ベースの事前準備」の効率化を実現しています。さらに、自動生成したソース、設計書をインプットにテストケース・テストデータの作成などにも活用しています。

Justware における生成 AI 活用パターン

Hitachi GenAI System Development Frameworkとの連携

HITACHI

パターン 2：システム開発の全工程に生成AIを適用する場合

- Justwareのインプットとなる詳細設計書をHitachi GenAI System Development Framework(新FW)を使って生成するケース
- 自動生成したソース、設計書をインプットにテストケース・テストデータの作成も新FWにて支援
- 本ケース以外も各工程での生成AIを活用した開発支援機能を有している



©Hitachi, Ltd. 2025. All rights reserved

上述した内容以外にも既に複数のユースケースでの活用が進められており、使用されている生成 AI もユースケースによって異なります。多いのは Azure OpenAI であり、LLM モデルとしては GPT-4o が最も多く、他に OpenAI o1 や OpenAI o1-mini を採用しているユースケースもあります。使用する生成 AI を限定していないのは、Part 1 でも述べたように、ツールの選択は各プロジェクトの判断で行うべきだと考えているからです。

あるユースケースにおいては、自然言語による推論を得意とする OpenAI o1 が向いている。ただし、OpenAI o1 の推論には時間がかかり、GPT-4o に比べて応答速度が遅いため、STEM 分野 (科学、技術、工学、数学) で OpenAI o1 と同等の性能を持ちながら処理速度を向上させた OpenAI o1-mini や、速度を重視するためにあえて GPT-4o を選択する、というアプローチも有効だと考えられます。

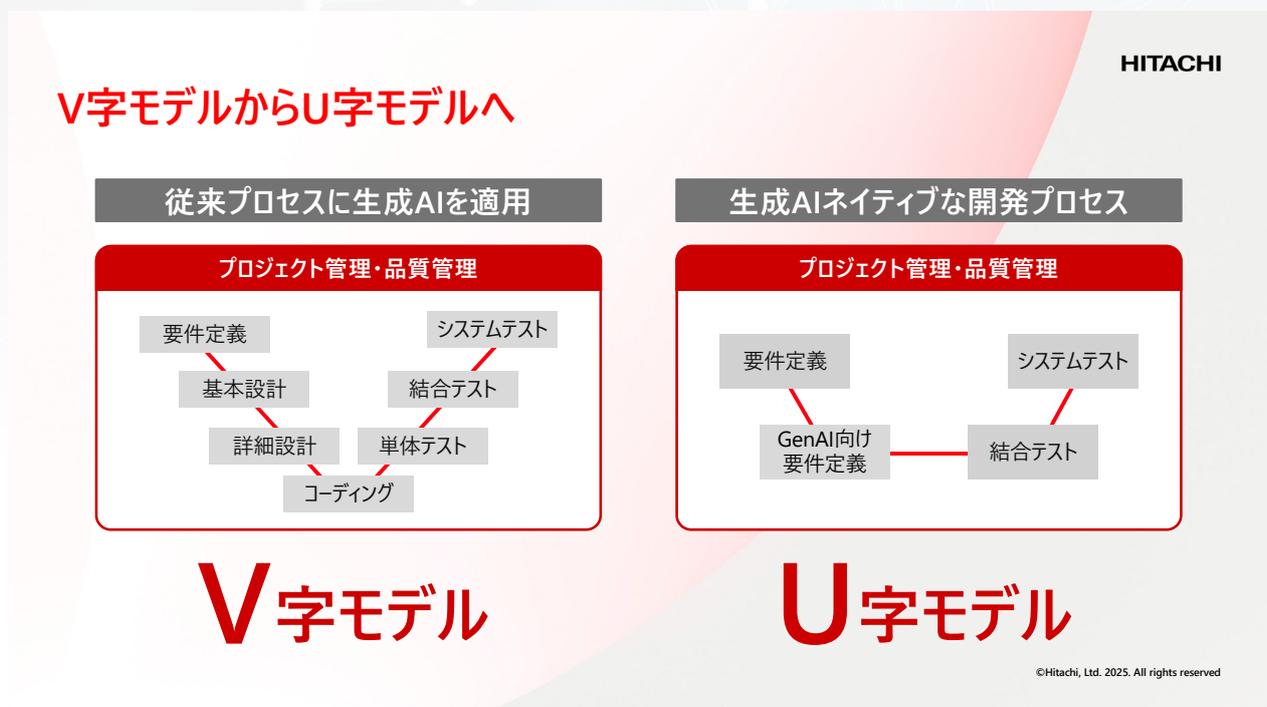
それではパターン 2 の取り組みは、どれだけの効果を期待できるのでしょうか。ユースケースによっても異なりますが、あるケースではコーディングの作業工数が 6 割以上削減されています。

なお、パターン 2 を実現するためのツールを活用したシステム開発への生成 AI 適用支援サービスについては、2024 年 5 月から一般提供を開始しています。

2-4 今後の展望

ここまで紹介してきた 2 つのパターンでの生成 AI 活用は、Justware OSS ベースの開発プロセスにおいて、コーディングや単体テスト (一部の結合テスト) といった「製造工程」を中心に効率化しつつ、システム開発の全工程を対象に取り組む内容です。これらはお客様への提供を前提に進められており、いずれのパターンも 2024 年春に一般提供を開始しています。

さらに、製造工程を完全に生成 AI で自動化することで、以下のように従来の「V 字モデル」から「U 字モデル」へと、開発プロセスを抜本的に改革していく、といったことも研究しています。



このようなアプリケーション開発における取り組みに加えて、インフラ設計および実装領域での適用も進めつつあります。ここでは、システム設計書やシステム構成図の作成、システム設計書のレビュー、環境構築の手順書作成やそのチェックなどの作業が必要になります。

もちろん、カバーできているタスクはまだ限られています。今後は生成 AI 適用タスクを順次拡大し、最終的に「システム開発全てのプロセス」に生成 AI を適用していくことを目指しています。

日立製作所のシステム開発における生成 AI 活用 ～ GitHub Copilot 活用の取り組みを中心に ～

Part 2 システム開発プロセスへの生成 AI 活用

■ 監修



五十嵐 聡



溝江 彰人



斎藤 岳

日立製作所 アプリケーションサービス事業部 テクノロジートランスフォーメーション本部

生成AIシステムエンジニアリング部
部長・GenAIアンバサダー

LSH適用推進部
担当部長

LSH適用推進部
部長・GenAIアンバサダー

■ お問い合わせ

日本マイクロソフト株式会社
108-0075 東京都港区港南 2-16-3 品川グランドセントラルタワー

株式会社日立製作所
「システム開発への生成 AI 適用支援サービス」サイト
<https://www.hitachi.co.jp/products/it/appsvdiv/service/genai/>

- ・ Microsoft、Azure、Azure OpenAI は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。
- ・ GitHub、GitHub Copilot は、GitHub, Inc. の米国およびその他の国における登録商標または商標です。
- ・ その他記載の会社名、製品名などは、それぞれの会社の登録商標または商標です。
- ・ 本ホワイトペーパーの内容は執筆当時 (2025 年 4 月) のものであり、変更が生じている可能性があることをご了承ください。
- ・ 本ホワイトペーパーは情報提供のみを目的としております。明示的または暗示的を問わず、本ドキュメントにいかなる保証を与えるものではありません。