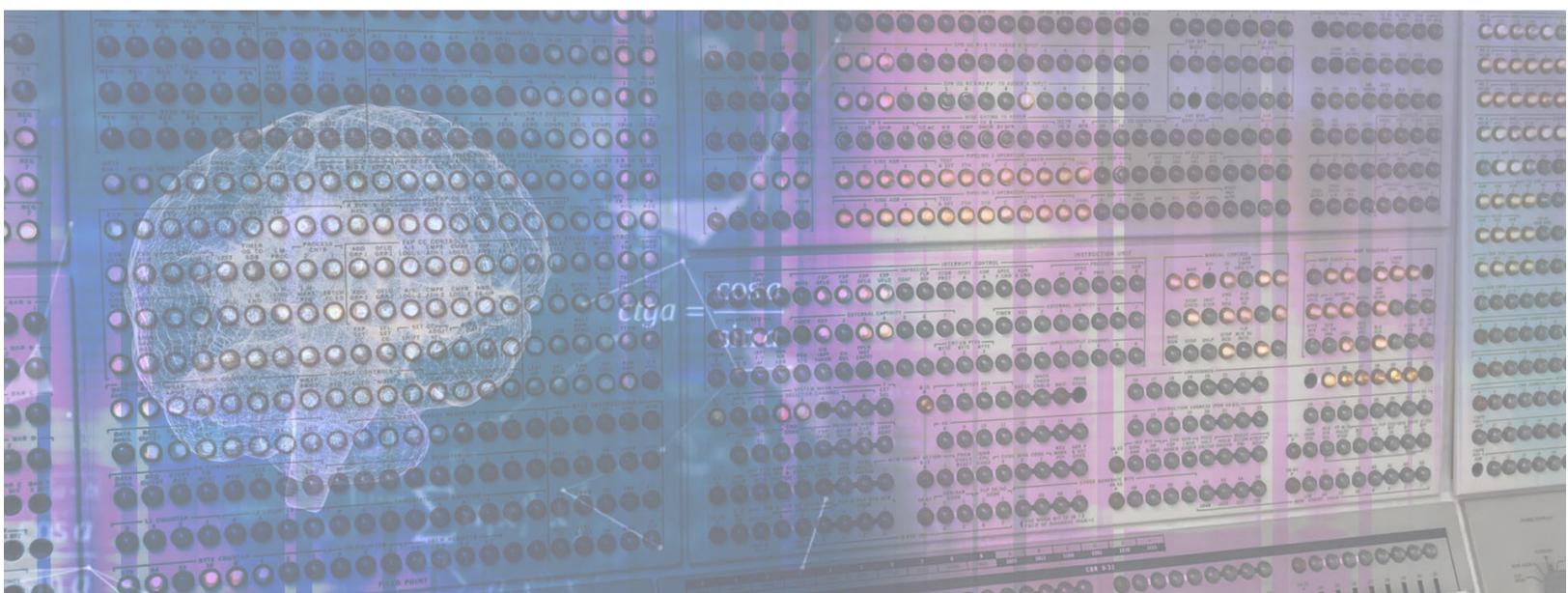




Intellyx™



AI をローコードアプリケーション開発に連携させる 3つの方法

SAP 向け Intellyx ホワイトペーパー

マネージングパートナー、ジェイソン・ブルームバーグ (Jason Bloomberg) 著

2023 年 8 月



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

ローコードのツールやプラットフォームは、アプリケーション構築の市場で成熟の域に達しています。今や企業は、こうしたプラットフォームを頼りにアプリケーション開発のコストと時間を削減し、開発者とビジネスステークホルダーとの連携を促進しています。

一方、人工知能 (AI) 市場は急激な混乱期を迎えており、ソフトウェアベンダーや顧客は ChatGPT のような生成 AI プラットフォームによる AI 主導の熱狂に巻き込まれています。

それどころか、ローコードの分野が生成 AI によって崩壊するかもしれません。生成 AI なら、言葉で指示するだけである程度使えるソースコードを生成できるからです。では、生成 AI はローコードに取って代わるのでしょうか。

実際には、崩壊に至るほどの混乱は生じないと思われます。実現の可能性が高いものとしては、既存のローコードプラットフォームへの AI の組み込み、AI によるローコードアプリケーション構築の強化、そして、AI モデルとそれに関連するトレーニングデータセットの構築、トレーニング、最適化という困難な作業へのローコード原則の適用が挙げられます。

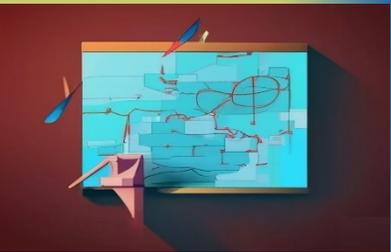


ホワイトペーパー：AI をローコード
アプリケーション開発に連携させる 3 つの方法

AI とローコードは、互いに敵か味方か？

ローコードツールは、ほかのどのテクノロジーにもまして、誰もが知るビジネスと IT の分断を解消してきました。ビジネスユーザーは簡単にソフトウェアアプリケーションのコントロールやインプットができるようになり、専門開発者はより効率的かつ顧客を重視したソフトウェア開発が可能になりました。

ローコード（およびその同類であるノーコード）により、エンジニアはソフトウェア開発の重労働から解放され、ビジネスステークホルダーと協働しながらデジタルオフリングを作成できるようになりました。しかも、ハンドコーディングはほとんど必要ありません。



ローコード（およびその同類であるノーコード）により、エンジニアはソフトウェア開発の重労働から解放され、ビジネスステークホルダーと協働しながらデジタルオフリングを作成できるようになりました。しかも、ハンドコーディングはほとんど必要ありません。

同時に、こうしたツールにより、事業部門の市民開発者（コーディングのスキルを持たないビジネスユーザー）は、アプリケーションの構築に挑戦してさまざまなレベルで精密なアプリケーションを作成できるようになり、IT 部門に依頼するのはテストや統合、ガバナンスといった難易度の高い課題だけになります。

市場には多種多様なローコードツールがあり、機能もさまざまですが、一般的にこうしたツールはアプリケーションビルダー向けのビジュアルインターフェースを備え、アプリケーションを構成するためのドラッグ & ドロップ機能とシンプルなウィザードが組み合わされています。



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

ローコードツールの仕組みは、2 つの基本的なパターンに従っています。

1 つは、コードジェネレーターです。アプリケーションビルダーがローコードインターフェースのボタンを押すと、最新のソースコードで構築された実行中のアプリケーションが現れます。このアプリケーションは、ビルダーが選択する実行環境（オンプレミスまたはクラウド）で動作します。

もう 1 つは、宣言型のプラットフォームです。ローコードインターフェースの視覚的なアプリケーションが動的に解釈され、基盤となるプラットフォーム（通常はクラウド）上で自動的に実行されます。

ローコード市場は、2023 年初頭まで全般的にこうした状況でした。しかしそれも、ChatGPT のような大規模言語モデル (LLM) をベースとした生成 AI プラットフォームの登場で一変しました。



まもなく人々は、生成 AI プラットフォームが簡単なプロンプトからソースコードを生成できることを知りました。生み出されるアプリケーションは完璧とは言えないものの、手作業でコーディングする時間を劇的に短縮できます。アプリケーションビルダーは、自分たちが望むアプリケーションの詳細を記述するだけでよいのです。

まもなく人々は、生成 AI プラットフォームが簡単なプロンプトからソースコードを生成できることを知りました。生み出されるアプリケーションは完璧とは言えないものの、手作業でコーディングする時間を劇的に短縮できます。アプリケーションビルダーは、自分たちが望むアプリケーションの詳細を英語（あるいは別の言語でも可）で記述するだけでよいのです。



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

今日の生成 AI のパワーとその急激なイノベーション曲線を目の当たりにすれば、ローコードはそのランドスケープもろとも AI 革命に屈してしまうと考えても無理はありません。

しかし、事実はずっと複雑です。AI とローコードの関係は確かに急激に進化していますが、この 2 つの破壊的なテクノロジーの間には、さまざまな相乗効果に向けた明確なイノベーションのトレンドがすでにくつも存在しているのです。

イノベーション 1：ローコードを活用してアプリケーションやプロセスに AI 機能を組み込む

生成 AI は、現在最も注目されているもののひとつです。しかし実際には、AI という用語は、機械学習やディープラーニング、自然言語処理といったいくつもの技術を包括しており、その多くは何年も前から存在しています。

こうした技術はどれも、急激な革新のさなかにあります。それに加えて大げさな誇大広告や憶測、さらには時折発生する不正行為によって混迷が深まっているため、意思決定者にとっては健全な AI 戦略の策定が困難になっています。

とはいえ、画像認識、光学式文字認識 (OCR)、文書処理、自然言語翻訳、チャットボットなど、成熟した信頼性の高い多種多様な AI 機能があるのも事実です。

こうした機能は通常、ローコードを活用するアプリケーションビルダーにとっては、アプリケーションビルダーがアプリケーションに導入できるモジュール型のアプリケーション構成部品となります。

例えば、住宅ローンを処理するアプリケーションに OCR ステップが搭載されていれば、撮影された文書を PDF として読み取り、検索可能なテキストに変換できます。ローコードアプリケーション内で AI が果たすもう 1 つのよくある役割は、自然言語翻訳のステップです。Google 翻訳を使用するのに似ていますが、こちらはアプリケーションのワークフローに組み込まれます。



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

AI は、アプリケーション固有の意思決定を支援することができます。例えば機械学習は、IT システムの性能の問題や金融サービスのワークフローに内在する潜在的な不正など、データセットの異常を発見するのに特に役立ちます。

特定した異常に基づいて、アプリケーションはローコードアプリケーションのロジックに従った修復ワークフローやその他のプロセスを実行できます。

イノベーション 2：AI でローコードアプリケーション開発を強化

ワークフローやアプリケーションへの AI の組み込みは、ローコード AI の一部にすぎません。おそらく、もっと強力なイノベーションと言えるのは、アプリケーション開発のプロセスそのものの支援に AI を活用することです。



「次の最適なアクション」ツールは、実質的にはローコードワークフローのためのオートコンプリートです。例えば、e コマースのプロセスでは、クレジットカード情報の入力要求が次のステップであることを、AI が過去のデータに基づいて予測します。

「次の最適なアクション」ツールは数年前から利用されています。この AI ベースの機能は、実質的にはローコードワークフローのためのオートコンプリート機能です。例えば、e コマースのプロセスでは、クレジットカード情報の入力要求が次のステップであることを、AI が過去のデータに基づいて予測します。

オートコンプリートに似たこの機能は、データに対して機能するだけでなく、データ出力の自動フォーマット、適切なグラフやチャートの推奨、データ自体のクリーニングに対しても有効です。



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

例えば、売上レポートには、たいてい共通の要素とフォーマットがあります。売上レポートをローコードアプリケーションで生成する必要がある場合、過去の同様のレポートに基づいて AI がフォーマットしてくれます。

今日の生成 AI プラットフォームのコード生成機能も、アプリケーションコンポーネントとアプリケーション全体のどちらに対しても、このローコードパターンに従います。

その名が示すように、ローコードでは多少のハンドコーディングが可能です。一般的には、特別な注意を要する固有のタスクに対するもので、カスタムユーザーインターフェースウィジェット、レガシーシステムへの統合、固有のビジネスロジックなどがあります。

生成 AI は、こうしたユースケースをすべて取り除き、ローコードアプリケーションビルダーがアプリケーションに直接組み込める作業コードを生成します。

生成 AI が成熟するにつれ、まったく新しいローコード方式が確立されていきます。

現在、ほとんどのローコードツールが従っているのが、「箱と線」によるフローチャートベースの方式か、構成に重点を置くウィザード方式（またはその 2 つの組み合わせ）のどちらかです。

生成 AI がもたらす 3 つ目の方式では、アプリケーションビルダーが AI にアプリケーションの詳細を過不足なく伝えると、その条件を満たすアプリケーションを AI が構築してくれます。生成 AI は単にソースコードを生成するのではなく、宣言型ローコードプラットフォームが直接解釈できる構成ベースの記述を作成するのです。

現在、ローコードに適したアプリケーションの多くは、e コマース、プロジェクト管理、登録リスト、タイムシートなど、さまざまなカテゴリーに分類されます。そのため、現在市場で見られるローコードプラットフォームには、たいてい、こうしたお馴染みのアプリケーションのユースケースに対応する数多くのテンプレートが提供されており、開発者は自由に変更できるようになっています。

生成 AI は、アプリケーションテンプレートのアイデアをさらに進め、テンプレートだけでは実現できないカスタマイズやほかの固有のアプリケーション要件をサポートします。これ



ホワイトペーパー：AI をローコード
アプリケーション開発に連携させる 3 つの方法

により、特定のビジネスニーズを満たすためにテンプレートを調整する時間と労力を削減できます。

イノベーション 3：ローコードの原則を活用した AI モデルの構築、 反復、展開

前述した 2 つのイノベーションが示唆するよりも、さらに広範なパラダイムシフトが起きています。現実的に、AI はアプリケーションの本質を変えようとしているのです。

AI アプリケーションは、コードや宣言的な設定ベースのロジックに依存するのではなく、モデルに依存します。AI モデルとは、大規模なデータセットのデータを処理したり分析したりするためにコンピューターをトレーニングするのに使用するツールやアルゴリズムのことです。

そのため、結果として得られるアプリケーションの動作は、モデルのトレーニングや基礎となるデータセットに依存します。アプリケーションの動作を直接構築しようとする人間の労力に依存するものではありません。

その代わりに、人間の役割は、アプリケーションを構築するという従来の概念から、モデルやデータセットの調整やデータセットがモデルをトレーニングする方法の調整にシフトします。

現在、こうした活動の難易度は高いため、データサイエンティストや AI エキスパートの専門知識が必要です。

まず、問題となるのはデータです。組織は既存のデータから価値を引き出したいと考えます。しかし難題は、モデルがデータにアクセスして理解できるように情報を組み立てることです。

今日の組織では、ほとんどの情報が非構造的、つまり、PDF ファイル、メール、ワープロ文書、ビジネスフォームなど、人間が作成した成果物で構成されています。従来は、こうした成果物に含まれる多くのデータ要素を人間が手作業でタグ付けし、AI モデルが理解できるようにする必要がありました。

ローコードツールは、こうしたタグ付けや一般的なデータクリーンアップの課題を劇的に簡素化します。データのクリーニング、クロップ、構造化、タグ付けをポイントアンドクリック



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

クで行えるのです。しかも、ローコードツールは、タグ付けプロセスを自動化しつつ、人的エラーを修正してくれます。

また、AI モデルの最適化はトライアンドエラーをベースにしているため、何度も繰り返す必要があります。こうしたトライアンドエラーは、通常、AI モデルをデプロイした後も、要件の進化とソースデータの追加に伴って継続されます。

継続的にモデルを最適化するというこの課題は、次世代のローコードプラットフォームにとっては絶好の機会となります。第一世代のローコードプラットフォームがハンドコーディングの複雑さと専門知識の必要性を軽減したのに対し、新世代のローコードプラットフォームでは、より多くの AI アプリケーション作成者が AI モデルとそれに関連するトレーニングデータセットを構築、最適化できるようになるのです。



第一世代のローコードプラットフォームがハンドコーディングの複雑さと専門知識の必要性を軽減したのに対し、新世代のローコードプラットフォームでは、より多くの AI アプリケーション作成者が AI モデルとそれに関連するトレーニングデータセットを構築、最適化できるようになります。

モデルの構築と最適化にローコードを適用することで、ローコードがアプリケーション開発の際に実現したのと同じように、人間にメリットをもたらします。ローコードでは、経験豊富な人材がミスを減らしながら迅速に作業できるため、プロジェクトをいち早く進めることができます。その結果、ほかのプロジェクトに移行する時間も確保できるようになります。

一方、経験の浅い人もローコードによってこうした活動が行えるようになり、また、データアナリスト、技術志向のビジネススペシャリスト、特定分野の専門家などは AI ベースのアプリケーションを独自に開発し、成熟させることができるようになります。



ホワイトペーパー：AI をローコード アプリケーション開発に連携させる 3 つの方法

こうしたローコードベースの AI モデル構築におけるビジョンはまだ大部分が将来に向けたものですが、注目すべきはそのようなツールを現在の市場に投入しているベンダーの例があることです。こうした最先端の製品は、通常、不正検知、IoT ベースの推論、その他の専門分野など、特定の適用領域に分類されます。

しかし、ローコードのコアであるコラボレーション、使いやすさ、迅速な市場投入といったメリットが活かされれば、やや懸念はあるものの、ローコードと AI の組み合わせによって AI ベースのアプリケーション構築が普及する日は近いはずで

Intellyx の見解

AI とローコードは、急速なイノベーションのさなかにあります。そのため、特に生成 AI の成熟度が高まるにつれて、イノベーションはさらに急激に進化すると予想されます。

市場カテゴリーとしてのローコードは成熟の停滞期に入っていましたが、そこに生成 AI がインパクトを与えたのです（多くの市場セグメントも同様でした）。この破壊の先に予測できるのは、ベンダーが最高の AI 駆動型ローコード製品の開発でしのぎを削ることで生まれる急速なイノベーションです。

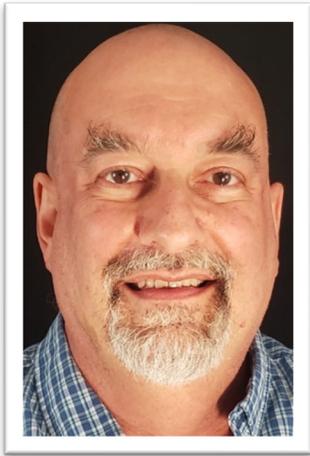
アーリーアダプターがリスクを負ってでも新技術に飛びついてくるため、一部の企業はこうした破壊を優位に活用できます。とはいえ、大半の企業は、アーリーアダプターによって市場が淘汰されるのを待つのが賢明です。

早期採用のリスクが自社のビジネスに適さないのであれば、リスクの少ない道を選ぶのは不名誉なことではありません。現在市場に出回っている成熟したローコード製品を活用しつつ、AI 戦略を強化すればいいのです。AI 駆動型のローコードが成熟すれば、そうしたプラットフォームを活用する機会はいくらでも訪れるはずで



ホワイトペーパー：AI をローコード
アプリケーション開発に連携させる 3 つの方法

著者について



ジェイソン・ブルームバーグ (Jason Bloomberg) 氏は、エンタープライズ IT 産業の分析企業である Intellyx の創業者兼マネージングパートナーです。IT 業界を代表するアナリスト、著者、基調講演者であり、エンタープライズテクノロジーとデジタルトランスフォーメーションにおけるさまざまな破壊的トレンドに関する世界的に著名な専門家です。

Thinkers 360 による [Top 50 Global Thought Leaders on Cloud Computing 2023](#) で第 13 位、[Top 50 Global Thought Leaders on Mobility 2023](#) で第 10 位に選出されています。また、Analytica の 2022 年 [Who's Who in Cloud?](#) の主要なソーシャルアンプリファイア（影響力のある情報発信者）であり、Team leadersHum の [Top 50 Agile Leaders of 2022](#) にも選出されています。

ブルームバーグ氏は、2019 年 10 月に出版された『[Low-Code for Dummies](#)』を含む 5 冊の著者または共著者です。

SAP Build について

SAP Build は、ローコードによるアプリケーション開発と自動化に必要な要素のすべてを提供します。エンタープライズアプリケーションの作成、プロセスの自動化、ビジネスサイトの設計を、ドラッグ & ドロップ操作で簡単に行えます。SAP Build を使用することで、すべての企業資産を可視化し、シームレスに統合することができます。また、最新のアプリケーション開発では、IT 部門とビジネス部門が垣根を越えて安全にコラボレーションできるようになります。SAP Build の詳細と無料体験については、www.SAP.com/Build をご覧ください。

Copyright © Intellyx LLC. 本稿の最終的な編集権は Intellyx に帰属します。本稿の執筆に AI は使用されていません。画像クレジット：[Craiyon](#) にて自動生成。