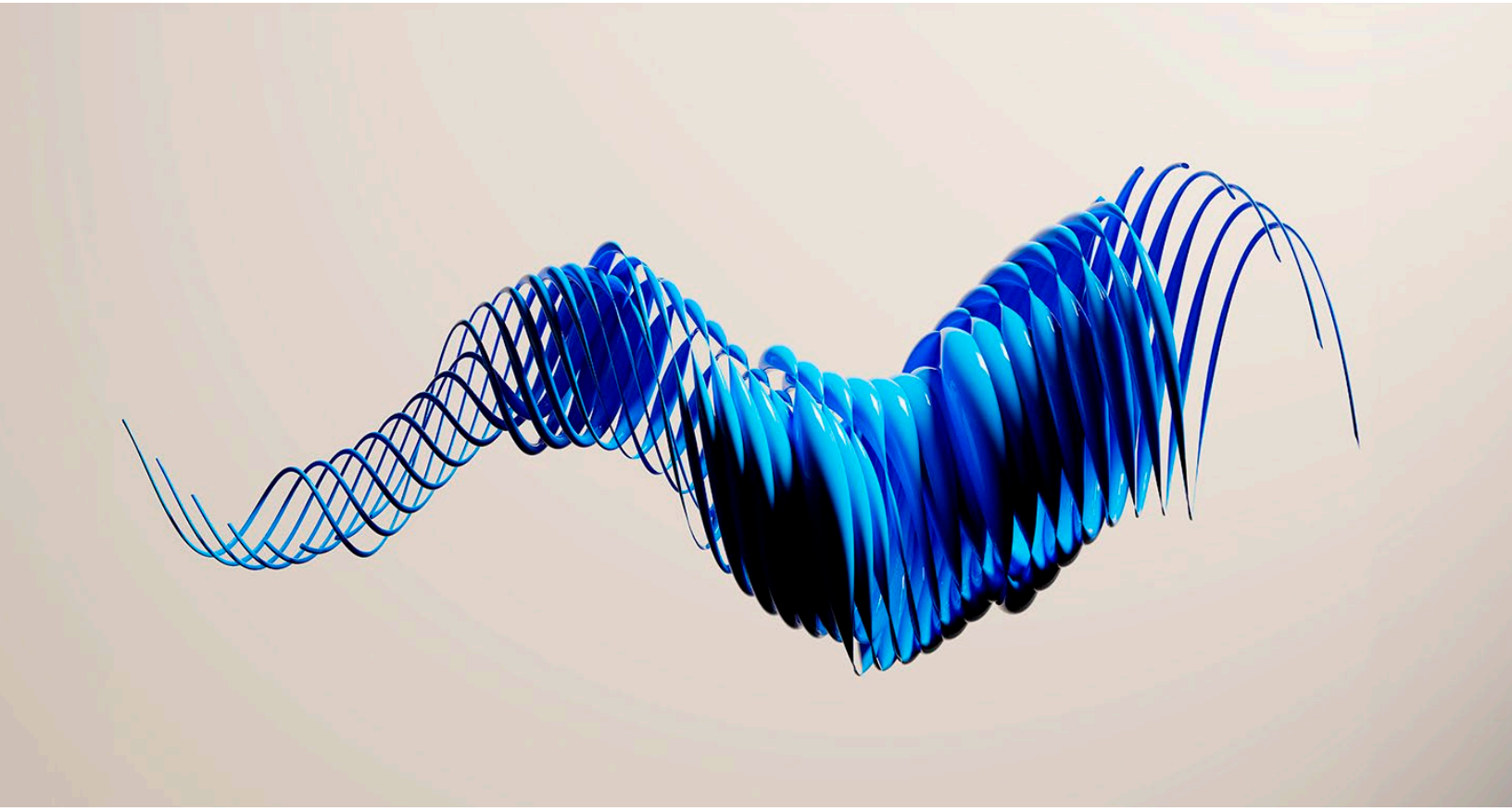
McKinsey Digital

生成AI時代のテクノロジー：CIOとCTOへの指南書

生成AIの活用によるビジネス部門と技術部門の再考に向けて、CIOとCTOがとるべき9つのアクション

本記事はAamer Baig、Sven Blumberg、Eva Li、Douglas Merrill、Adi Pradhan、Megha Sinha、Alexander Sukharevky、Stephen Xuによる共著であり、マッキンゼー・デジタルの見解を代表するものである。



2023年7月

日本語版掲載によせて

日本で失われた30年が言われて久しい。世界の先進諸国が堅調で健全なインフレ率を維持し、先端技術の恩恵を受けながらデジタルトランスフォーメーションに成功し、その経済恩恵を受けている一方で、日本はスイスのビジネススクールIMDの2023年「世界競争力ランキング」で35位と、過去最低の順位となった。日本最強と言われた1989年以降、1992年まで維持したランキング1位は、もはや遠い昔の話となっている。投下資本効率性や労働生産性においても、日本は欧米と比べて一部のセクターを除いてほぼ壊滅的に低調・横ばいである。

一方、明るいニュースも生成AIの分野から出てきた。マッキンゼー・グローバル・インスティテュート(MGI)の2023年の試算によれば、生成AIがもたらす世界経済インパクトは、あと数年で日本のGDPに匹敵する規模に達すると言われている。政府与党から人工知能プロジェクトチーム(PT)によるホワイトペーパーが発行され、G7でも首脳陣によりその重要性が確認された。特に日本において生成AIにより投下資本効率性や労働生産性の改善が見込まれるのは、マーケティングやソフトウェアエンジニアなどの領域だ。

その適用リスクはもちろんある。しかし、やらないことのリスクが大きい今、むしろ何もしないことの方がリスクになるという認識が、経営幹部の間で日々増大していることも事実だ。生成AIのリスクを理解し、先手を打って荒波を乗り切り、先行者利益を得るか否かは、経営幹部の手腕にかかっていると言っても過言ではない。本稿が、日本のCIOやCTO、その他の経営層の意思決定の判断として、有益な情報となり、将来の経済成長の第一歩を踏み出す材料として寄与することを願ってやまない。

工藤 卓哉(パートナー、QuantumBlack、AI by McKinsey共同リーダー)

連日のように、ビジネスを根本から変革するような、生成AI関連の新たな開発に関する話題がメディアをにぎわせている。このような熱狂が巻き起こるのも当然のことといえる – マッキンゼーの調査によれば、生成AIは年間2.6〜4.4兆ドル相当の価値を生み出す可能性を秘めているのである[1]。

CIO（Chief Information Officer: 最高情報責任者）およびCTO（Chief Technology Officer: 最高技術責任者）は、その経済価値を獲得する上で重要な役割を担っているが、まずは以前にも同じような状況にあったことを思い起こす必要がある。インターネット、携帯電話、SNSなどの新たなテクノロジーやツールが登場した際には様々な実験やパイロット（試験）運用を繰り返したにもかかわらず、結局、実際には大きなビジネス価値を捕捉することは困難であることが分かったケースも多かった。これらの取り組みから学んだ教訓は、今でもそのほとんどを生かすことができ、特にパイロット運用から本格運用にまで規模を拡大することに関して大いに参考となる。CIOおよびCTO（以下、「CIO/CTO」）にとって、生成AIの可能性を持続可能なビジネス価値に昇華すべく経営陣を導くにあたり、生成AIブームがこれらの教訓を応用する絶好の機会となっている。

我々は、数十人の技術リーダーと対話を行い、またマッキンゼーを含む50社以上で実施している生成AIの取り組みを分析することにより、生成AIによる価値創造、テクノロジーとデータの統合、ソリューションの拡張、リスク管理に向け、あらゆる技術リーダーに実践することを勧めたい9つのアクションを特定した（コラム「主な用語について」を参照）。

コラム

主な用語について

生成AIは、機械学習（ML）技術を用いて大量の（公開）データでトレーニングを行い、学習したパターンを活用して新たなコンテンツ（テキスト、コード、画像、動画）を生成できるAI（人工知能）の一種である。

基盤モデル（Foundation Model）：大量のラベルなし・非構造化データで訓練されたディープラーニングモデルで、そのまま幅広いタスクに活用できる。また、ファインチューニングにより、特定のタスクにも適応させることができる。例として、GPT-4、PaLM 2、DALL·E 2、Stable Diffusionなどが挙げられる

大規模言語モデル（LLM: Large Language Model）：基盤モデルの種別の一つ。非構造化テキストデータを大量に処理し、単語や単語の一部（トークン）の間の関係を学習することができる。これにより、LLMは自然言語テキストを生成し、情報の要約や抽出といったタスクを実行できる。LLMの例として、Cohere社のCommand、またグーグルが開発した対話型AIサービスのBardに搭載されているLaMDA（ラムダ）などが挙げられる

ファインチューニング：特定のタスクでモデルのパフォーマンスを向上させるために、訓練済みの基盤モデルを微調整すること。事前訓練に使用した膨大な量のデータセットとは異なり、少量のラベル付きデータセットを用いて短期間で学習させる。これにより、少量のデータセットに隠されたニュアンスや専門用語、特定のパターンをモデルが学習し、適応することができる

プロンプトエンジニアリング：生成AIモデルが期待通りの（正確な）アウトプットを生成するように、プロンプト（生成AIへの指示、入力文）を設計、改良、最適化するためのプロセス

[1] "The economic potential of generative AI: The next productivity frontier（日本語版：生成AIがもたらす潜在的な経済効果：生産性の次なるフロンティア）" マッキンゼー（2023年6月14日）

1. **生成AIの導入に関して自社の姿勢を**速やかに**決定**し、従業員に対して具体的な取り組みに関するコミュニケーションを行い、生成AIへの適切なアクセス権を付与する。

2. ビジネス部門を見直し、**生産性向上、成長、新たなビジネスモデルを通じて価値を創造するユースケースを特定する**。生成AIの真のコストとリターンの見積もりに必要な「財務AI（FinAI）」能力を構築する。

3. **技術部門を見直し**、ソフトウェア開発に必要な生成AI能力を早急に構築し、技術的負債の削減およびITオペレーションにおける手作業の大幅な削減に注力する。

4. **既存のサービスを利用するか、オープンソースの生成AIモデルを適応**させて独自の機能を開発するかを判断する（独自の生成AIモデルを構築・運用するには、少なくとも当面は数千万ドルから数億ドルのコストを要する）。

5. **エンタープライズ・テクノロジー・アーキテクチャ（ETA）をアップグレード**して、生成AIモデルを統合・管理する。モデルが相互に、また既存のAIや機械学習（ML）モデル、アプリケーション、データソースと連携できるよう調整する。

6. 構造化データおよび非構造化データの両方に対応することで、**高品質なデータへのアクセスを実現するデータアーキテクチャを構築**する。

7. 製品チームとアプリケーションチームに承認済みのモデルをオンデマンドで提供できるよう、**一元的に取り組みを行う機能横断型の生成AIプラットフォームチームを組成**する。

8. ソフトウェアエンジニア、データエンジニア、MLOpsエンジニア、セキュリティ専門家など、主要な役割のスキルアップに投資する。しかし、生成AIが与える影響は様々であるため、**職務や熟練度によってトレーニングプログラムを調整**する必要がある。

9. **新たなリスク展望を評価**し、モデル、データ、ポリシーに関する**継続的なリスク軽減策を確立**する。

1. 生成AIの導入に関する自社の姿勢を明らかにする

生成AIの利用がますます拡大するにつれ、リスク低減のためにCIOやCTOが従業員に対して一般公開されているアプリケーションの利用を禁止する動きが見られるようになった。このような企業はイノベーションの機会を逃してしまうおそれがあり、従業員からは、重要な新しいスキルを構築する機会が奪われていると捉えられている場合もある。

したがって、完全に禁止してしまうのではなく、CIO/CTOはリスクやコンプライアンス関連のリーダーと連携してリスク軽減の真の必要性と、自社内で生成AIのスキルを構築することの重要性とのバランスを見極める必要がある。特に生成AI領域においては、各国によって、その対応方針や法規制のあり方が大きく異なり始めており、そうした動向も注意深く見据える必要がある。そのためには、ビジネスが許容できるリスクレベルや、生成AIが自社の全体的な戦略にどのように適合するかについて合意を形成し、生成AIに関する自社の姿勢を確立することが求められる。このステップにより、企業は全社的な方針やガイドラインを迅速に決定できるようになる。

方針が明確に定義されたら、CIO/CTOは適切なアクセス権限を設定するとともにユーザーにとって分かりやすいガイドラインを作成し、各リーダーが組織全体に共有する。企業によっては、生成AIに関する全社的なコミュニケーションを展開したり、特定のユーザーグループに対して生成AIの幅広いアクセス権を提供したりしているところもある。また、ユーザーが社内データをアプリケーションに入力すると即時に警告するポップアップを作成したり、一般公開されている生成AIのサービスにアクセスするたびにガイドラインが表示されるようにモデルを設定したりしている企業もある。

2. 生産性向上、成長、新たなビジネスモデルを通じて価値を創造するユースケースを特定する

CIO/CTOは、これまでに多数の企業で見られた「企画倒れ」を阻止する役割を果たすことが求められる。大きな役割として、例えばCEOやCFO、およびその他のビジネスリーダーと連携して、生成AIが既存のビジネスモデルに与える影響や、新たなビジネスモデ

ルや価値の源泉を生み出す方法を熟考することなどが挙げられる。CIO/CTOは、技術的な可能性に関する専門知識を活用し、企業全体で生成AIのメリットを享受できる最も価値のある機会と課題、およびメリットを享受できないものを特定する必要がある。場合によっては、生成AIが最良の選択肢ではないこともある。

例えば、マッキンゼーの調査によると、生成AIはマーケティング関連の特定のユースケース（顧客選好を探る抽象的な非構造化データの分析など）で約10%、顧客対応関連のユースケース（高機能な生成AIボットなど）で最大40%、生産性を向上させることができる[2]。生成AIの価値を最大限まで引き出すことを目指してユースケースの最適な分類方法を検討する際に、CIO/CTOは特に重要な役割を担っている。分類する方法としては、ユースケースをドメイン別（カスタマージャーニー、ビジネスプロセスなど）に分ける方法、または種類別（クリエイティブコンテンツの作成、バーチャルエキスパートなど）に分ける方法などが挙げられる。世の中には生成AIのユースケースがすでに多数存在するため、機会を特定すること自体はそれほど戦略的なタスクとはいえない。しかし、当初は人材や必要な能力が限定的であることを考慮すると、自社が生成AIに関する優先順位を決定する際に、CIO/CTOが実現可能性およびリソースの見積もりを提供することが大きな支援となる。

このレベルの助言を行うには、技術リーダーはビジネス部門と協働して、生成AIの取り組みに関する真のコストとリターンの見積もりに必要なFinAI能力を構築しなければならない。コスト計算は特に複雑となる可能性がある。なぜなら、ユニットエコノミクス（1単位当たりの経済性・採算性）は複数のモデルやベンダーのコスト、モデルの相互作用（クエリが複数のモデルからの入力を必要とする場合があり、それぞれ使用料がかかる）、継続的な使用料、人間の監視コストを考慮する必要があるからである。

3. 技術部門の見直しを行う

生成AIは、技術部門の業務を大幅に変革する可能性がある。CIO/CTOは、生成AIが技術部門のあらゆる分野に与える潜在的な影響を総合的に検討する必要があるが、経験と専門知識を蓄積するために早急に行動を起こすことが重要となる。CIO/CTOが最初に注力すべき領域は、以下の3つとなる：

- **ソフトウェア開発：**マッキンゼーの調査よると、生成AIのコード生成支援によりソフトウェアエンジニアの作業時間を短縮でき、コード開発を35〜45%、コードのリファクタリングを20〜30%、開発関連のドキュメント作成を45〜50%、それぞれ高速化することができる[3]。また、生成AIはテストプロセスを自動化し、エッジケースのシミュレーションも行えるため、チームはリリース前により堅牢なソフトウェアを開発し、（生成AIにコードベースに関する質問をすることなどで）新たな開発者のオンボーディングを加速することができる。このような機能を活用するには、大規模なトレーニング（「アクション8」を参照）と、急増するコード量を管理できるよう、DevSecOps（開発・セキュリティ・運用）プラクティスにより統合パイプラインとデプロイパイプラインを自動化する必要がある。
- **技術的負債**：技術的負債はIT予算の20〜40%を占め、開発ペースを著しく遅らせる要因となる[4]。CIO/CTOは、技術的負債のバランスシートを見直し、生成AIの機能（コードのリファクタリングや翻訳、テストケースの自動生成など）をどのように活用すれば技術的負債の削減を促進できるかについて検討すべきである。
- **ITオペレーション（ITOps）**：CIO/CTOは、ITOpsの生産性向上の取り組みを見直し、生成AIがどのような形でプロセスを加速できるかを判断する必要がある。生成AIは、特に次のようなタスクで有効に活用できる：（1）パスワードのリセット、ステータスのリクエスト、セルフサービスのエージェントによる基本的な診断などのタスクの自動化、（2）ルーティングの改善によるトリアージと解決の迅速化、（3）テーマや優先度などの面で重要な内容を洗い出し、それに対する適切な回答の生成、（4）大量のログを分析することでオブザーバビリティ（可観測性）を向上し、特に注意を要する事象の検出、（5）標準作業手順書、インシデント報

[2] 前掲レポート
[3] Begum Karaci Deniz, Martin Harrysson, Alharith Hussin, and Shivam Srivastava, "Unleashing developer productivity with generative AI" マッキンゼー（2023年6月27日）
[4] Vishal Dalal, Krish Krishnakanthan, Björn Münstermann, and Rob Patenge, "Tech debt: Reclaiming tech equity" マッキンゼー（2020年10月6日）

告書、パフォーマンスレポートなどの文書の作成。

4. 既存のサービスを利用するか、またはオープンソースの生成AIモデルを適応させるかを判断する

生成AIをどのような形で活用すべきかを模索する場合にも、従来から行ってきた「レンタル、購入、構築」のいずれを選択すべきかの判断が必要となる。ここでも基本的なルールは同じである：企業は、自社独自の優位性を生み出す生成AIの能力に投資し、一般的な領域に関しては既存のサービスを利用すべきである。

ここでは、これらの選択肢を以下の3つの類型に置き換えて検討する。

- **Taker**：チャットインターフェースやAPIを通じて一般公開されているモデルを利用し、カスタマイズはほとんど、または全く行わない。例えば、コード生成用の市販のソリューション（GitHub Copilotなど）や、デザイナーを支援する画像生成・編集用のソリューション（Adobe Fireflyなど）が挙げられる。これは、エンジニアリングおよびインフラストラクチャの両面で最もシンプルな類型で、一般的に最速で実装・運用が可能となる。これらは、基本的にプロンプトの形式でデータを公開モデルに入力する汎用的な類型となる。
- **Shaper**：よりカスタマイズされた結果を生成するため、モデルを社内のデータやシステムと統合する。例えば、生成AIツールを顧客関係管理（CRM）や財務システムに接続し、顧客の過去の販売履歴やエンゲージメントの履歴を組み込むことで、営業チームをサポートするモデルなどである。また、社内文書やチャット履歴を活用してモデルをファインチューニング（微調整）し、顧客担当者のアシスタントとして機能させることもできる。生成AIの機能を拡張して、より独自性の高い機能の開発、およびより高度なセキュリティやコンプライアンスの要件を満たすことを求めている企業には、Shaperが適している。

この類型では、生成AIモデルとデータを統合する方法として、一般的に2つのアプローチがある。一つは、「モデルをデータに投入する」方法で、モデルは組織のインフラ（オンプレミスまたはクラウド環境）でホストされる。例えば、Cohere社は、顧客企業のクラウドインフラ上に基盤モデルを展開しており、データ転送の必要性を低減している。もう一つのアプローチは、「データをモデルに投入」する方法で、組織はデータを集約し、大規模なモデルのコピーをクラウドインフラ上に展開するものである。どちらのアプローチも基盤モデルへのアクセスを確保するものであり、どちらを選択するかは、タスクやプロジェクトの性質・規模、および保有する各種リソースなどによって判断する。

- **Maker**：特定のビジネスケースに対応するため、基盤モデルを新たに構築する。基盤モデルの構築は複雑かつ高額で、膨大な量のデータ、深い専門知識、大規模な計算能力が必要となる。この選択肢では、モデルの作成および訓練に数千万ドルから数億ドルという多額の投資が必要となる。コストは、トレーニングのインフラ、モデルアーキテクチャ、モデルのパラメータ数、データサイズ、専門家のリソースなど、様々な要因により異なる。

技術リーダーが考慮すべきコストは、それぞれの類型で異なる（図表1）。モデルトレーニングの効率化やGPU（グラフィックス・プロセッシング・ユニット）による計算コストの低減などの新たな開発によりコストは低下しているものの、「Maker」は本質的に複雑であるため、短期的にこれを採用する組織は少数とみられる。ほとんどの組織は、汎用のソリューションを迅速に活用できる「Taker」と、基盤モデルの上に独自の機能を構築する「Shaper」との組み合わせを検討することになると考えられる。

5. エンタープライズ・テクノロジー・アーキテクチャ（ETA）をアップグレードして、生成AIモデルを統合・管理する

組織は、規模、複雑度、能力が異なる様々な生成AIモデルを使用することになる。価値を生み出すには、これらのモデルが相互に連携し、また組織の既存のシステムやアプリケーションとも連携可能な状態にする必要がある。そのため、生成AI用に個別の技術スタックを構築すること、複雑度が増す結果となる。一例として、ここでは顧客が旅行会社に予約の変更を求めるケースを検討してみる（図表2）。生成AIモデルが顧客に対応するには、複数のアプリケーションとデータソースにアクセスする必要がある。

図表1

各類型の総所有コスト（TCO）の見積もり

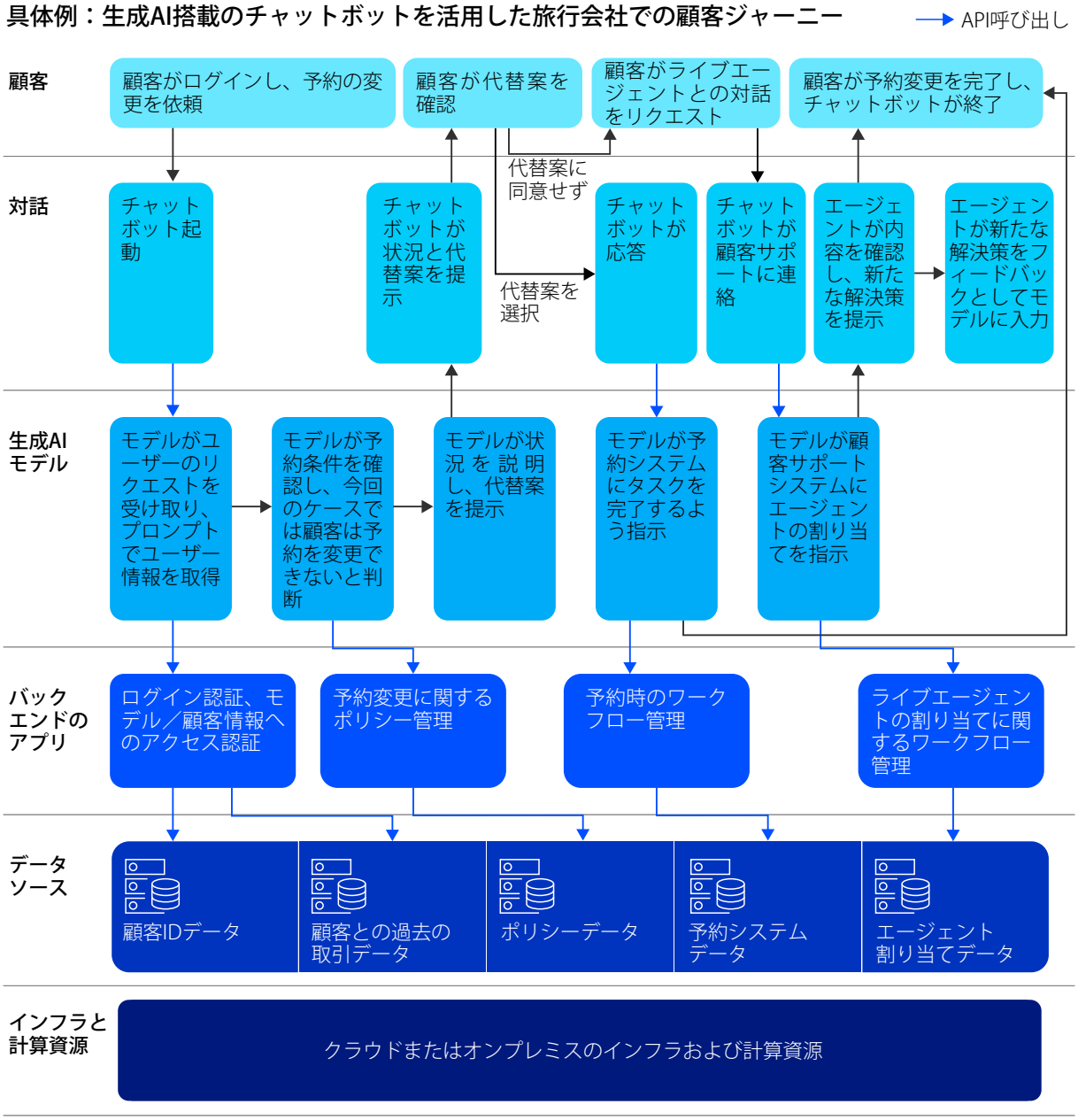
類型	ユースケース例	総所有コストの概算
Taker	– ソフトウェア開発者向け、既製のコード生成支援ツール – 汎用の顧客サービス用チャットボット（プロンプトエンジニアリングおよびテキスト形式のチャットのみ）	**50～200万ドル（初期費用）** – 既製のコード生成支援ツール：統合費用として～50万ドル。コストには6人編成のチームに要する3～4カ月分の人件費を含む – 汎用の顧客サービス用チャットボット：サードパーティ製のAPI上にプラグインレイヤーを構築する場合は～200万ドル。コストには8人編成のチームに要する9カ月分の人件費を含む **年間約50万ドル（運用コスト）** – モデルの推論： • 既製のコード生成支援ツール：日々1,000人のユーザーに対して年間～20万ドル。 • 汎用の顧客サービス用チャットボット：日々1,000人の顧客とチャットし、1回のチャットにつき1万トークンと仮定した場合、年間～20万ドル – プラグインレイヤーのメンテナンス：開発コストの10%と仮定した場合、年間最大20万ドル
Shaper	– 業界に特化したナレッジおよびチャット履歴でファインチューニングした顧客サービス用チャットボット	**200～1,000万ドル（初期費用、ただし後にモデルをファインチューニングする場合は別途必要）** – データおよびモデルのパイプライン構築：～50万ドル。コストには、5～6人の機械学習エンジニアとデータエンジニアが16～20週間かけて行うデータの収集・ラベル付けと、データのETL[1]に要する人件費を含む – モデルのファインチューニング[2]：1回のトレーニング当たり約10～600万ドル[3] • 低い方のコスト：コストには、計算コストと2人のデータサイエンティストに要する2カ月分の人件費を含む • 高い方のコスト：クローズドソースモデルに基づきファインチューニングしたコスト – プラグインレイヤーの構築：約100～300万ドル。コストには6～8人編成のチームに要する6～12カ月分の人件費を含む **年間約50～100万ドル（運用コスト）** – モデルの推論：年間最大50万ドル（運用コスト）。音声とテキストの両方で日々1,000件のチャットを行うと仮定 – モデルのメンテナンス：～50万ドル。MLOpsプラットフォーム[4]および1人の機械学習エンジニアが業務時間の50～100%をモデルのパフォーマンスの監視に費やした場合、年間10～25万ドルと仮定 – プラグインレイヤーのメンテナンス：開発コストの10%と仮定した場合、年間最大30万ドル（運用コスト）
Maker	– 医師の診断支援としてトレーニングした基盤モデル	**500万～2億ドル（初期費用、ただし後にモデルをファインチューニングする場合は別途必要）** – モデル開発：～50万ドル。コストには、4人のデータサイエンティストが既存の研究に基づき、3～4カ月をかけて行うモデルの設計、開発、評価に要する人件費を含む – データおよびモデルのパイプライン構築：約50～100万ドル。コストには、6～8人の機械学習エンジニアとデータエンジニアが約12週間かけて行うデータの収集と、データのETL[1]に要する人件費を含む – モデルのトレーニング[5]：1回のトレーニング当たり約400万～2億ドル[3]。コストには、計算コストおよび4～6人のデータサイエンティストに要する3～6カ月分の人件費を含む – プラグインレイヤーの構築：約100～300万ドル。コストには6～8人編成のチームに要する6～12カ月分の人件費を含む **年間約100～500万ドル（運用コスト）** – モデルの推論：1,000ユーザー当たり、年間約10～100万ドル。各医師が1日に20～25人の患者を診察し、1人の患者の診察時間を6～25分と仮定 – モデルのメンテナンス：年間約100～400万ドル（運用コスト）。MLOpsプラットフォーム[4]および3～5人の機械学習エンジニアによるモデルのパフォーマンスの監視に要する人件費を年間25万ドルと仮定 – プラグインレイヤーのメンテナンス：開発コストの10%と仮定した場合、年間最大30万ドル（運用コスト）

注記：エンジニアリングの最適化により、生成AIに関するコストは急激に変化しており、ここに提示した見積りは2023年半ば時点の総所有コスト（リソース、モデルのトレーニングなど）に基づく概算値となることに留意されたい

1. Extract（抽出）、Transform（変換）、Load（書き出し）
2. モデルは、約10万ページにおよぶ業界の専門的なドキュメントと、約1,000人の顧客担当者の5年間にわたるチャット履歴で構成するデータセットに基づきファインチューニングを実施（約480億トークン）。下限コストは、1%のパラメータをオープンソースモデル（LLaMAなど）で再トレーニングした場合で、上限コストはクローズドソースモデルで再トレーニングした場合。チャットボットへはテキストと音声の両方でアクセス可能
3. モデルは、各回のトレーニング後にハイパーパラメータ、データセット、モデルアーキテクチャに基づき最適化。必要に応じてモデルを定期的に更新（新たなデータを入手した際など）
4. Gilad Shaham, "Build or buy your MLOps platform: Main considerations," LinkedIn（2021年11月3日）
5. モデルは、650億～1兆のパラメータと、1.2～2.4兆個のトークンでトレーニング。ツールへはテキストと音声の両方でアクセス可能

図表2

生成AIを主要なタッチポイントで統合して顧客ジャーニーをカスタマイズする



Takerでは、このレベルの調整を行う必要ない。一方で、生成AIの機能拡張を求めるShaperやMakerでは、CIO/CTOはテクノロジーアーキテクチャ（TA）をアップグレードする必要がある。主な目的は、生成AIモデルを社内システムや企業アプリケーションに統合し、様々データソースへのパイプラインを構築することである。最終的に、生成AI機能の統合と拡張において鍵を握るのは、エンタープライズ・テクノロジー・アーキテクチャの成熟度である。

最近の統合・オーケストレーション[5]フレームワーク（LangChainやLlamaIndexなど）の進歩により、様々な生成AIモデルと他のアプリケーションやデータソースとの接続に要する労力が大幅に削減されている。また、統合についてもいくつかのパターンが出現しており、例えば、モデルがユーザーのクエリに応答する際にAPIを呼び出せるようにするものや（例として、GPT-4はFunction Calling「関数呼び出し」に対応）、ユーザーのクエリの一部として外部のデータセットからコンテキストデータを提供できるようにするもの（RAG: リトリーバル・オーグメンテッド・ジェネレーション）などが挙げられる。技術リーダーは、自組織の参照アーキテクチャおよび標準的な統合パターン（例えば、ユーザーやAPIを呼び出すモデルを識別する標準的なAPIのフォーマットやパラメータなど）を定義する必要がある。

生成AIを効果的に統合するには、以下の5つの主な要素をテクノロジーアーキテクチャに組み込む必要がある（図表3）：

- **コンテキスト管理とキャッシュ**：企業のデータソースからモデルに関連情報を提供する。適切なタイミングで関連するデータにアクセスすることで、モデルはコンテキスト（文脈）を理解し、質の高いアウトプットを生成できるようになる。キャッシュはよくある質問に対する回答を保存する機能で、これにより迅速な回答が可能となり、コストも低減できる。
- **ポリシー管理**：企業のデータ資産への不正アクセスを防止する。この管理により、例えば、従業員の詳細な給与・報酬情報を含む人事部門のAI生成モデルには、組織の他の部門からアクセスできないようにできる。
- **モデルハブ**：オンデマンドで提供可能なトレーニング済みおよび承認済みのモデルを保存し、モデルのチェックポイント、重み、パラメータのリポジトリとして機能する（Hugging Faceなど）。
- **プロンプトライブラリ**：各生成AIモデルに最適化したプロンプト（命令文）を保存するライブラリで、モデルの更新に伴うプロンプトのバージョン管理も含まれる。
- **MLOpsプラットフォーム**：生成AIモデルの複雑性を考慮し、アップグレードしたMLOps機能を含むプラットフォーム。例えば、MLOpsパイプラインには、モデルのタスク固有のパフォーマンス（適切なナレッジを取得する能力など）を測定する機能を搭載しておく必要がある。

アーキテクチャを進化させるには、CIO/CTOは急拡大する生成AIのプロバイダーやツールのエコシステムを効率よくかじ取りしなければならない。クラウドプロバイダーは、大規模なハードウェアや基盤モデルへの広範なアクセスを提供するだけでなく、様々なサービスも提供し、その種類も増加している。一方、MLOpsやモデルハブのプロバイダーは、基盤モデルを適応させて本番環境への導入に必要なツール、テクノロジー、手法を提供したり、特定のタスクの実行に向け、基盤モデルの上に構築し、ユーザーが直接アクセスできるアプリケーションを提供したりしている。生成AIモデルを効果的に展開・運用できるよう、CIO/CTOはこれらの様々な機能をどのように組み立て、統合するのが最適かを十分に検討する必要がある。

6. 高品質なデータへのアクセスを実現するデータアーキテクチャを構築する

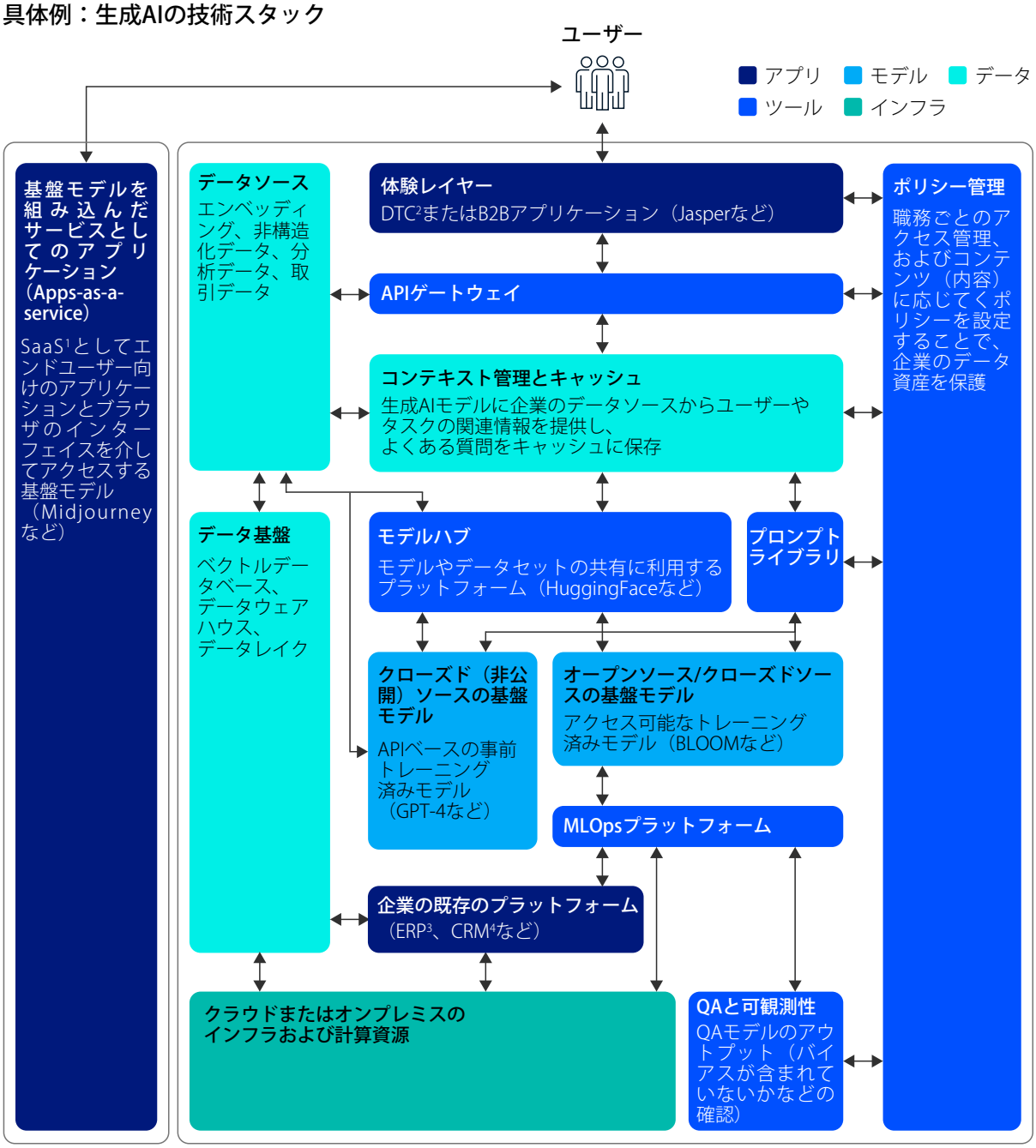
企業が生成AIモデルから価値を創出・拡大し、コスト削減およびデータやナレッジの保護体制の強化などを実現できるかは、自社のデータをいかに効率よく活用できるかにかかっている。その優位性を生み出すには、生成AIモデルと社内のデータソースとを接続するデータアーキテクチャが必要となる。これにより、モデルにコンテキストを提供したり、モデルのファインチューニングに役立てたりすることができるようになり、より適切なアウトプットの生成につながる。

[5] システムやアプリケーション、サービスなどの構築、設定、運用管理を自動化すること

図表3

生成AIに最適な技術スタックを構築する

具体例：生成AIの技術スタック



1. SoftwareasaService：サービスとしてのソフトウェア
2. Direct-to-Consumer：消費者への直接販売
3. EnterpriseResourcesPlanning：企業資源計画
4. CustomerRelationshipManagement：顧客関係管理

このような背景から、CIO、CTO、CDO（最高データ責任者）が緊密に連携し、以下のような取り組みを行うことが求められる：

- データを分類・整理し、生成AIモデルで利用可能な状態にする。技術リーダーは、構造化データおよび非構造化データの両方を含む包括的なデータアーキテクチャ[6]を構築する必要がある。これを実現するには、データを生成AI用に最適化する基準やガイドラインを整備しなければならない。例えば、多様性とサイズを向上させるために学習データを合成サンプルで補強する、メディアの種類を標準化したデータ形式に変換する、トレーサビリティとデータ品質を向上させるためにメタデータを追加する、データを更新する、などが挙げられる。
- 既存のインフラやクラウドサービスが、生成AIアプリケーションに必要な膨大な量のデータを保存および処理できるかを確認する。
- 生成AIモデルが「コンテキストを理解」できるよう、関連するデータソースに接続するデータパイプラインの構築を優先する。新たなアプローチとして、データを高次元ベクトルとして保存・取得できるベクトルデータベースの活用や、モデルに模範的な回答をいくつか例示し、より適切な回答を導く「Few-shot（数ショット）プロンプティング」のようなコンテキスト内学習（ICL）などが挙げられる。

7. 一元的に取り組みを担う機能横断型の生成AIプラットフォームチームを組成する

多くのテック企業は、プロダクト＆プラットフォーム型のオペレーティングモデルへの移行を進めている。CIO/CTOは、既存のインフラを基盤として、生成AIの機能をこのオペレーティングモデルに統合し、生成AIの迅速な導入拡大を推進すべきといえる。最初のステップは、生成AIのプラットフォームチームを立ち上げることである。このチームの主な目的は、製品チームやアプリケーションチームに対して、承認済みの生成AIモデルをオンデマンドで提供できるプラットフォームサービスの開発と保守である。また、このプラットフォームチームは、生成AIモデルをいかにして社内システム、企業アプリケーション、ツールと統合するかについてのプロトコルを定義し、責任あるAIのフレームワークなど、リスク管理に向けたて標準化した取り組みを策定・展開する。

CIO/CTOは、プラットフォームチームに適切なスキルを持つ人材を配属する責務も担い、統括者として上級技術リーダーを確保する。このチームを構成するのは、主に次のメンバーである：生成AIモデルを既存のシステム、アプリケーション、ツールに統合する「ソフトウェアエンジニア」、モデルを記録やデータソースを格納する様々なシステムに接続するパイプラインを構築する「データエンジニア」、モデルの選択とプロンプトを設計する「データサイエンティスト」、複数のモデルおよびモデルのバージョンのデプロイの管理と監視を行う「MLOpsエンジニア」、新たなデータソースでモデルをファインチューニングする「MLエンジニア」、データ漏えい、アクセス制御、アウトプットの精度、バイアス（偏見、先入観）などのセキュリティ問題を管理する「リスク専門家」。プラットフォームチームの実際の構成は、企業全体で対応するユースケースによって異なる。顧客向けのチャットボットを作成するような場合は、製品管理とユーザー体験（UX）に深い専門知識を有する人材が必要となる。

現実的には、プラットフォームチームはまず優先度の高いユースケースに対象を絞って取り組み、再利用可能な機能を構築したり、最適な方法を学んだりしながら、徐々に対象範囲を拡大していくことが望ましい。技術リーダーは、ビジネスリーダーと緊密に連携して、どのビジネスケースに資金や支援を提供すべきかを評価することが求められる。

8. 職務と熟練度に応じてスキル向上プログラムをカスタマイズする

生成AIは、従業員の生産性を大幅に向上し、能力を強化する可能性がある。しかし、享受できる恩恵は役割やスキルレベルによって異なるため、リーダーは従業員が実際に必要とするスキルをどのように構築すべきかについて再考する必要がある。

[6] Sven Blumberg, Jorge Machado, Henning Soller, and Asin Tavakoli, "Breaking through data-architecture gridlock to scale AI" マッキンゼー（2021年1月26日）

例えば、生成AIツールの一つであるGitHub Copilotを使用したマッキンゼーの最新の実証研究では、ソフトウェアエンジニアがコードを書く速度を35〜45%向上させることができた[7]。ところが、その効果は熟練のエンジニアと、経験が浅いエンジニアとの間で異なる結果となった。前者では50〜80%の向上が見られたが、後者では7〜10%速度が低下した。これは、生成AIツールのアウトプットについて、エンジニアがコードを評価、検証、改善する必要があり、経験の浅いソフトウェアエンジニアにとってはそれが困難だからである。逆に、顧客サービスのような技術的な要素が少ない仕事では、スキルの低い従業員が大いに生成AIの恩恵を受けており、ある調査によれば、生産性が14%向上し、離職率も低下したという事例がある[8]。

このような格差は、技術リーダーがCHRO（Chief Human Resource Officer: 最高人事責任者）と連携し、将来の労働力の構築に向けて人材マネジメント戦略を再考する必要性を改めて浮き彫りにしている。優秀な生成AIのトップ人材を採用することが重要となり、さらにそのような人材の希少性と戦略的重要性が増していることを考慮すると、技術リーダーは競争力のある給与モデルを設定したり、自社にとって重要な戦略的業務に携わる機会を提供したりするなど、離職防止の取り組みも実践すべきである。

技術リーダーの役割は、人材確保だけにはとどまらない。既存のほぼすべての職務が生成AIの影響を受けるため、職務、熟練度、ビジネス目標に基づいて、どのようなスキルが必要かを明確に把握した上で、従業員のスキル向上に重点を置く必要がある。例として、ソフトウェアエンジニアを取り上げてみる。初心者を対象としたトレーニングでは、コード生成に加え、優れたコードレビュアーとなれるよう導くことに重点を置くべきである。一般的な文章で執筆と編集で違いがあるように、コードのレビューにも生成とは異なるスキルセットが必要となる。

ソフトウェアエンジニアは、優れたコードとはどのようなものかを理解し、生成AIが作成したコードの機能性、複雑性、品質、可読性をレビューし、脆弱性がないかを確認する一方で、自身がコードに品質やセキュリティの問題を持ち込まないように留意する必要がある。さらに、ソフトウェアエンジニアは、これまでとは異なる観点でコードを作成することが求められ、生成AIツールからより適切な回答を得られるように、ユーザーの意図をよく理解した上でプロンプトを生成し、コンテキストデータを定義する必要がある。

CIO/CTOは、テック人材の育成だけではなく、技術系以外の人材に生成AIのスキルを構築する際にも重要な役割を果たす。メール作成やタスク管理などの基本的な業務に生成AIツールを活用する方法を理解するだけでなく、生産性や成果物の品質を向上できるよう、すべての従業員が様々な機能を使いこなせるようになることが望ましい。そのために必要なトレーニングや認定精度を提供すべく、CIO/CTOが社内のアカデミー制度の調整・構築を支援することが求められる。

経験の浅いエンジニアの価値が下がることで、初級レベルの人材が最も多い従来のピラミッド型の構造から、技術職の大部分を経験豊富な人材が占めるダイヤモンド型（ひし形）のような構造への移行が加速すると考えられる。具体的な取り組みとして、初級レベルの従業員のスキルを早急に向上させ、複雑度の低い手作業（ユニットテストの記述など）に専念する職務を削減していくことなどが挙げられる。

9. 新たなリスク展望を評価し、継続的なリスク軽減策を確立する

生成AIはメリットだけではなく、新たな倫理的な問題やリスクも伴っていることに留意する必要がある。例えば、事実とは異なる不適切な回答を提示する「幻覚（ハルシネーション）」、機密性の高い個人情報の偶発的な漏えい、モデルが使用する大規模なデータセットに内在するバイアス、知的財産権（IP）侵害のおそれなどである。CIO/CTOは、倫理、人道、コンプライアンスに関する問題に精通し、法律の文言（国によって異なる）に従うだけではなく、責任を持って自社の評判を守るという精神にも従わなければならない。

この新たな状況に対処するには、サイバーセキュリティ対策の大幅な見直し、およびモデル開発を開始

[7] "Unleashing developer productivity with generative AI" マッキンゼー（2023年6月27日）
[8] Erik Brynjolfsson, Danielle Li, and Lindsey R. Raymond, Generative AI at work, National Bureau of Economic Research（NBER）working paper, number 31161（2023年4月）

する前にリスク評価およびリスク軽減策を特定して、ソフトウェア開発プロセスを更新する必要がある。これにより、問題の発生確率を低減し、プロセスに遅延が発生することを抑えることができる。幻覚に関する効果実証済みのリスク軽減策として、例えば次のようなものが挙げられる：モデルが回答を生成する際の創造性のレベル（「temperature」と呼ばれるパラメータ）を調整する、関連する社内データでモデルを補強して、より多くのコンテキストを提供する、生成する内容に「ガードレール（悪用を防ぐ対策）」を設けるライブラリを使用する、「モデレーション」モデル（不適切な回答の監視）を使用する、明確な免責事項を追加する。初期段階では、組織が不可避な挫折を克服して後に教訓として生かせるよう、失敗コストが低い分野に焦点を当てて生成AIのユースケースを展開すべきである。

データプライバシーを確立するには、機密データをタグ付けするプロトコルの設定・運用、異なる部門間のデータアクセス管理の設定（人事部門の報酬データなど）、データが社外で使用される場合の追加の保護措置、プライバシーに関するセーフガード（保護対策）の確立などが重要となる。例えば、アクセス制御のリスクを軽減するために、モデルにプロンプトが与えられた際に職務によってアクセスを制限する、ポリシー管理層を設定している組織もある。また、知的財産に関するリスクを軽減するには、CIO/CTOは基盤モデルのプロバイダーに対して、使用するデータセットのIP（データソース、ライセンス、所有権）の透明性を確保するよう求めることが重要である。

生成AIは、史上まれに見る速度で急成長している技術カテゴリーとして位置づけられている。技術リーダーは、生成AI戦略の定義と策定を不必要に遅らせるわけにはいかない。この分野は今後も急速な進化を続けるとみられるが、CIO/CTOは責任を持って、本稿で紹介した9つのアクションを参考として、効果的に生成AIの力を最大限活用いただきたい。

著者について

Aamer Baigはマッキンゼー・シカゴオフィスのシニアパートナー、**Sven Blumberg**はデュッセルドルフオフィスのシニアパートナー、**Eva Li**はベイエリアオフィス（サンフランシスコ）のコンサルタント、**Megha Sinha**はベイエリアオフィスのパートナー、**Douglas Merrill**は南カリフォルニアオフィスのパートナー、**Adi Pradhan**と**Stephen Xu**はトロントオフィスのアソシエイトパートナー、**Alexander Sukharevsky**はロンドンオフィスのシニアパートナー

日本語版
共著者、監修者
工藤 卓哉（パートナー、QuantumBlack、AI by McKinsey共同リーダー）
マッキンゼー 関西オフィス

監修者
川村 俊輔（プロダクト担当ディレクター）
マッキンゼー 東京オフィス

本稿の執筆にあたり、以下のマッキンゼーのメンバーから多大なる協力を得た。著者一同より、ここに感謝の意を表する：Stephanie Brauckmann、Anusha Dhasarathy、Martin Harrysson、Klemens Hjartar、Alharith Hussin、Naufal Khan、Sam Nie、Chandrasekhar Panda、Henning Soller、Nikhil Srinidhi、Asin Tavakoli、Niels Van der Wildt、Anna Wiesinger