



アプリケーション・モダナイゼーションの基本



アプリケーション・モダナイゼーションとは何なのか。なぜ、いま必要とされるのか。本ホワイトペーパーでは、その基本から実現に向けたロードマップ、ソリューション、アーキテクチャーの概要までを紹介する。

“2025年の崖”克服に向けたITシステムの方向性

今、ビジネスの最前線で話題に上っているのが「2025年の崖」という言葉である。経済産業省は、「デジタルトランスフォーメーションレポート～ITシステム『2025年の崖』の克服とDXの本格的な展開～」^{※1}を発表した。

これによると、多くの経営者が、将来の成長、競争力強化のために、新たなデジタル技術を活用して新たなビジネスモデルを創出し、柔軟に改変するデジタル・トランスフォーメーション（以下、DX）の必要性を理解しているという。

だが、既存システムの多くは、事業部門ごとに構築されて全社横断的なデータ活用ができなかったり、過剰なカスタマイズがなされてい

たりなどの理由で複雑化、ブラックボックス化している。たとえ経営者がDXを望んでも、課題を解決しなくてはならず、そのためには業務自体の見直し、すなわち経営改革を求められる。一方で、現場サイドの抵抗も大きく、いかにこれを実行するかが課題となっている。

こうした課題を克服できないと、DXが実現できないだけでなく、2025年以降、年間で最大12兆円、現在の約3倍の経済損失が生じる可能性がある。それが2025年の崖である。では、2025年の崖を、逆に飛躍のきっかけにするには何が必要なのか（図1）。

※1:ITシステム「2025年の崖」の克服とDXの本格的な展開（経済産業省）
https://www.meti.go.jp/shingikai/mono_info_service/digital_transformation/20180907_report.html

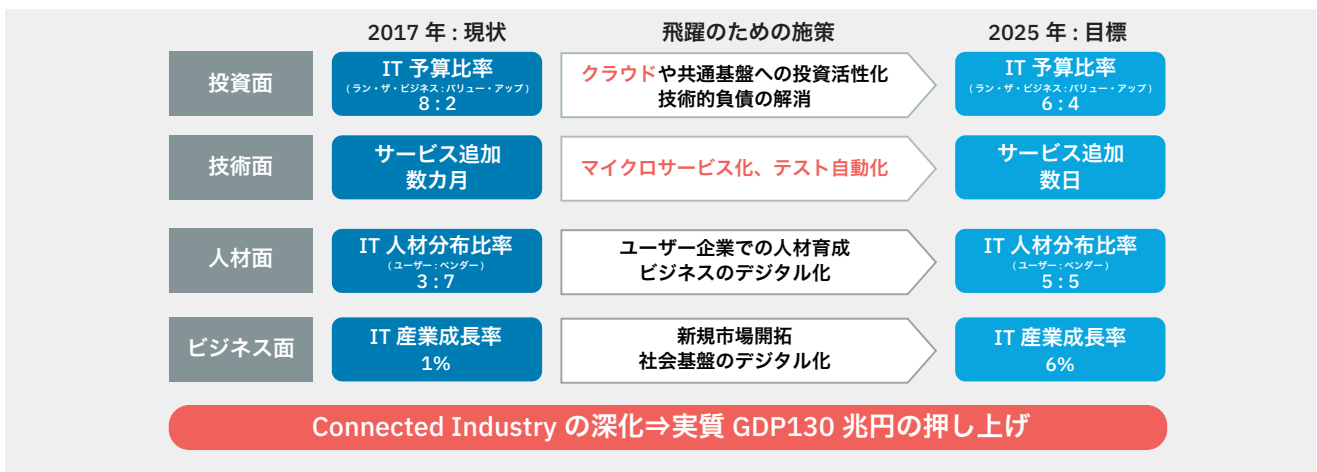


図1 2025年の崖を飛躍のきっかけにするには
(出典：経済産業省、デジタルトランスフォーメーションに向けた研究会 2018年9月7日発行 DXレポート～ITシステム「2025年の崖」の克服とDXの本格的な展開～)

アプリケーション・モダナイゼーションの進め方

2025年の崖を飛躍のきっかけにするために技術面で重要なのが、クラウド・ネイティブ・コンピューティングだ。そして、クラウドにネイティブなアプリケーションを実現するアプリケーション・モダナイゼーションの技術的要素が「マイクロサービス」である。DXの実現には、新たなアイデアや市場環境の変化をとらえビジネスにつなげるために、アプリケーションをいかに柔軟かつすばやく構築できるかが鍵となっている。これまでの基幹システムが苦手としていた変化への対応力が、マイクロサービスは非常に優れており、従来型のモノリシック（一枚岩）なシステムを、いかにマイクロサービス化するかがポイントとなるのである。

マイクロサービスとは、RestfulAPIなどを介し、小さな複数のサービスをつなぎ、1つの大きなサービスを構築する手法のことで、Uberが配車やGPSなど細かなサービスを

組み合わせて、短期間で米国のタクシー業界を震撼させる仕組みを構築したことで知られる。こうした事例が「破壊的イノベーション」として取り上げられ、その中核技術としてマイクロサービスが注目されている。

マイクロサービス化への具体的な戦略として、最も重要なのが「リフト&シフト」だ（図2）。ITリソースをまずはモノリシなままクラウド型へと移行（リフト）し、そこからさらに、クラウドの長所を最大限に生かせる分散型のクラウド・ネイティブ・アプリへと進化（シフト）させるやり方である。

クラウド・ネイティブ・アプリに移行できれば、クラウド基盤を最大限に活用することで、DevOpsを前提にした素早いアプリケーションの開発とリリース、柔軟なアプリケーションの変更といった自由度が格段に高まる。

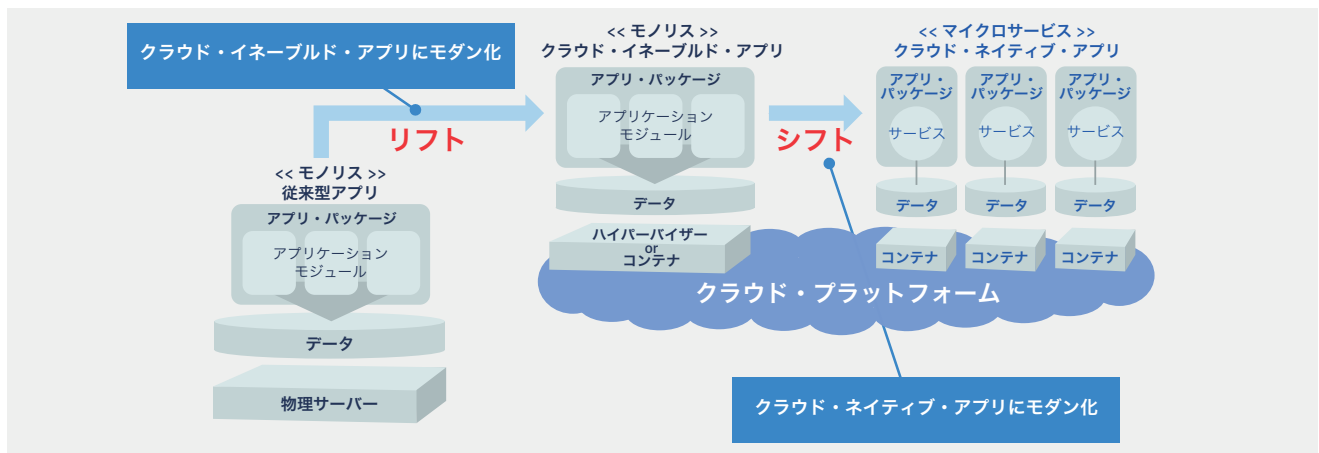


図2 リフト&シフトによるクラウド導入戦略

アプリケーション・モダナイゼーションの3つのパターン

クラウド・ネイティブ・アプリの狙いは、アプリケーションの保守を可能にするサービス化による、柔軟なモジュラー構造の実現にある。

従来型のモノリシな仕組みは、UI、ビジネスロジック、データベースアクセスといったライブラリー同士が密結合しているため、一度構築したアプリケーションの変更が難しい。一方で、マイクロサービスは各サービス間が疎結合であるため、ビジネス要件の変更があっても柔軟にアプリケーションを変えられるのがメリットだ（図3）。

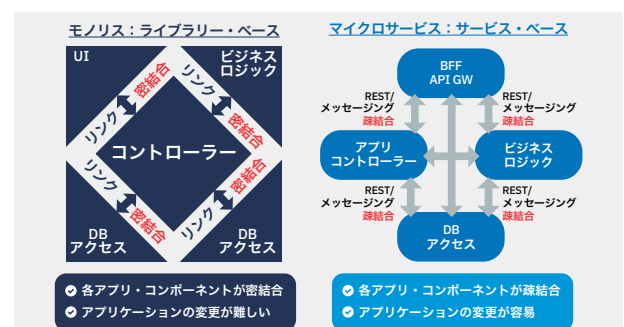


図3 マイクロサービスの概念

ここで、既存のアプリケーションをモダナイゼーションする際のパターンを見ていこう (図 4)。

最も取り組みやすいタイプ I は、統合型のデータベース上で、インフラをコンテナ・ベースに置き換えることでクラウド対応のアプリケーションにリフトするパターンだ。また、タイプ II ではその上のアプリケーションをマイクロサービスによって分散化させる。さらに、モダナイゼーションの最終形であるタイプ III では、データベースも分散設計にすることで、基盤も含めた分散化を可能にする。データベースの運用やスキーマ更新を柔軟化できるのは大きなメリットである一方、データベースの結果整合性を許容しなくてはならないなどの課題も出てくる。流れとしては、低リスクのタイプ I から、アプリケーションのモダナイズであるタイプ II、さらにデータベースの改革へと段階的に着手するのが一般的なパターンと言える。

タイプ III では、データベースの容易な変更のため、各サービスを介して分散データベースにアクセスするアーキテクチャーとなっている。複数のサービスが 1 つのデータベースに直接アクセスするようなパターンを取った場合、データベース・スキーマの変更が複数のサービス実装の変更に波及してしまう。これではデータベースの容易な変更が阻害される。そこで、データベース・スキーマ変更によるアプリケーション・ロジックへの影響を最小化するために、各データベースにはそれぞれ特定の 1 つのサービスを介してアクセスするようなアーキテクチャーを取っているのだ。

マイクロサービスでは、開発、運用の流れという観点で、ドメ

イン分析・設計、開発、運用という流れで組むことが多い。ICT アーキテクト、ソフトウェアエンジニア、SRE (Site Reliability Engineering) エンジニアが担うのは従来どおりだが、ドメイン分析・設計のフェーズを、ビジネス・スペシャリストが担当する点がこれまでとの大きな違いとなっている。

注意しておきたいのは、アプリケーション・モダナイゼーションのプラットフォームは画一的なものではないということだ。アプリケーションの変更が頻繁でなければ現状のままモノリスな仕組みを残し、アプリケーションの新規開発や既存アプリの作り直しのみマイクロサービス化するという方法が適しているケースもある。必ずしも、マイクロサービス化と分散データベースへの刷新がベストな選択というわけではないということだ。

■コンテナと Kubernetes

タイプ I の要であるコンテナは、アプリケーションの実行に必要な OS のライブラリーやランタイムをアプリケーションとともにパッケージ化し、開発、テスト、本番環境への簡単な移行や、高速起動、高速スケールを実現する。代表的なサービスとして Docker、コンテナの管理ツールとして Kubernetes が知られている。

仮想サーバーごとにゲスト OS の立ち上げが必要な従来型の仮想マシン (VM) よりも、コンテナの方がホスト可能なアプリケーションがずっと多いことや、処理が軽いこと、起動が非常に速いことが特徴だ。モノリスとクラウド・ネイティブ環境の双方向への展開が容易であることが、モダナイゼーションの中核技術になっていることの理由であろう。

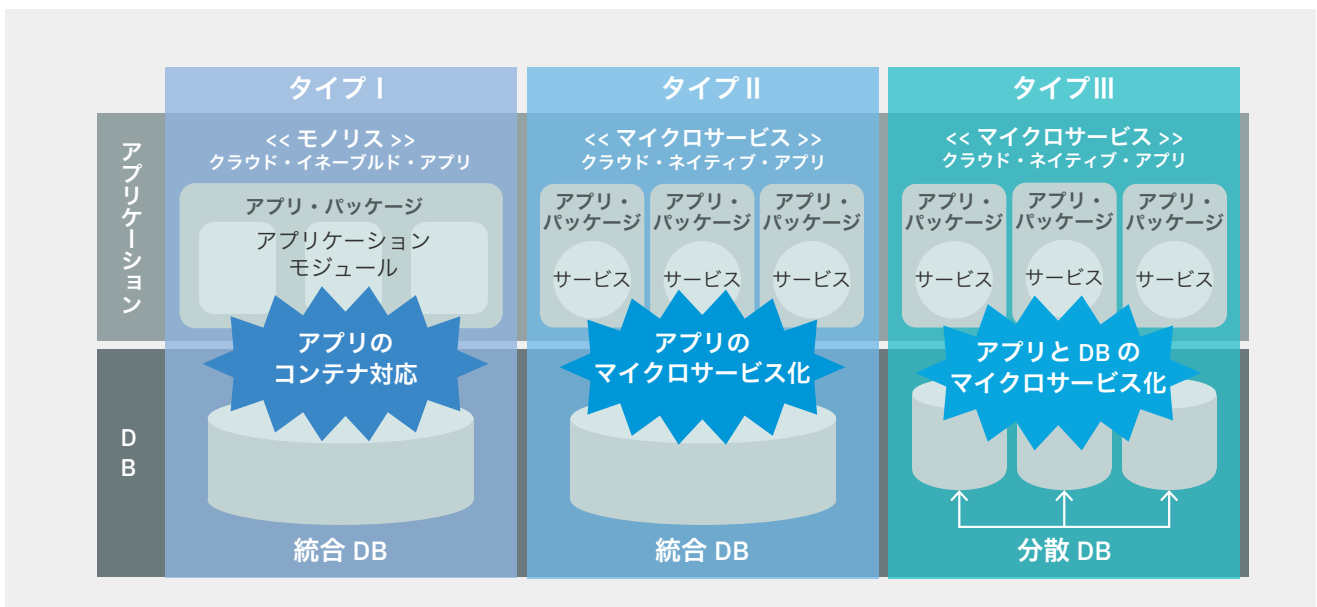


図 4 アプリケーション・モダナイゼーションの 3 つのパターン

■ DevOps の実現

ここまで見てきたように、超分散型ともいえるマイクロサービスのアーキテクチャーを実装しようとした際に、その開発と運用の体制を組むのに重要なキーワードが DevOps である。

マイクロサービスでは、ドメイン分析・設計、開発、運用というプロセスを繰り返すアプローチを取る。それを実行するのは、ビジネス、アプリ開発と基盤担当者で混成する 1 チーム体制である。この時、最も適している構築手法が DevOps の実現と切り離せないアジャイル・プロセスなのである。

具体的には、継続的デリバリーや継続的インテグレーション、ドメイン駆動設計、サイト・リライアビリティ・エンジニアリング、スクラムといった方法論を用いることが、DevOps の成功、ひいてはアプリケーション・モダナイゼーションの実現へとつながっていく。言うまでもなく、アプリケーション・モダナイゼーションによって実現する変化対応力の高いシステム環境は、デジタル化によってめまぐるしく移り変わるビジネス要件への柔軟な対応力につながる。すなわち、DX の原動力になるのだ。

プラットフォームを最適化する IBM のソリューション

2025 年の崖を控え、その対応が必要となる中で、ビジネスで勝ち抜くための最新のプラットフォームが必要になる。IBM では、「IBM Cloud Pak for Applications」や「IBM Cloud Pak for Integration」など、ユーザー企業をサポートする効果的な仕組みを提供している。これらを活用していくことで、アプリケーション・モダナイゼーションを効率的に進めていくことができる（図 5）。

IBM Cloud Pak for Applications は、クラウド・ネイティブ・アプリの作成や既存のアプリケーションの最新化を、数週間、数カ月

という単位ではなく、数日または数時間での変革プラットフォームである。オンプレミスのソフトウェアを活用することも、クラウドに最適化された次世代のデータとソフトウェアを簡単に統合することもできる。

また、IBM Cloud Pak for Integration は、シンプルで迅速、安全にシステムを統合するためのマルチクラウド・プラットフォームの実装を支援する。

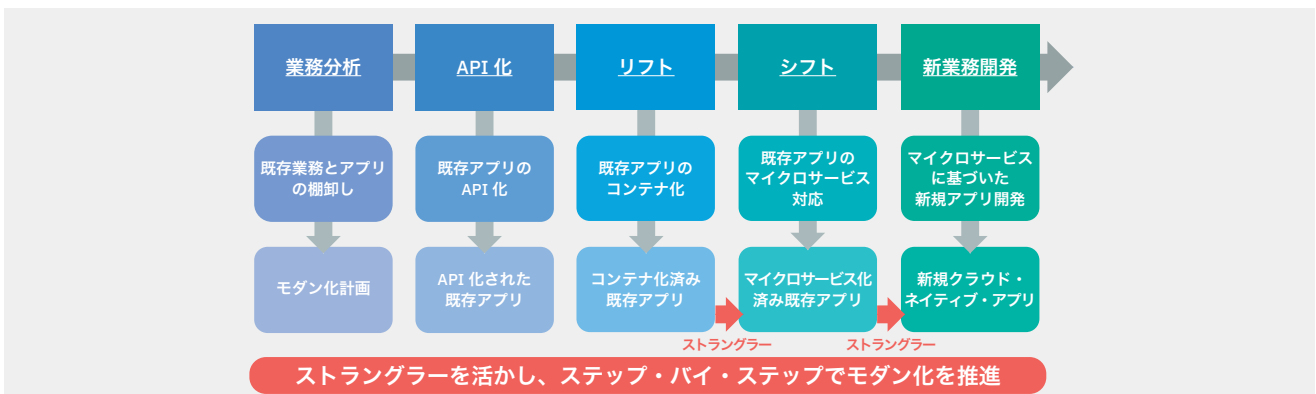


図 5 アプリケーション・モダナイゼーションのロードマップ例

関連サービス情報

<http://ibm.biz/appm-jp>

お問い合わせ

IBM アクセスセンター

☎ 0120-550-210

受付時間 9:00~17:00 (土、日、祝日を除く)