

企業システムの100年史と近未来

～柔軟性の高いアーキテクチャへの拡張～

野村證券株式会社

フロンティア・リサーチ部

長谷川 哲也

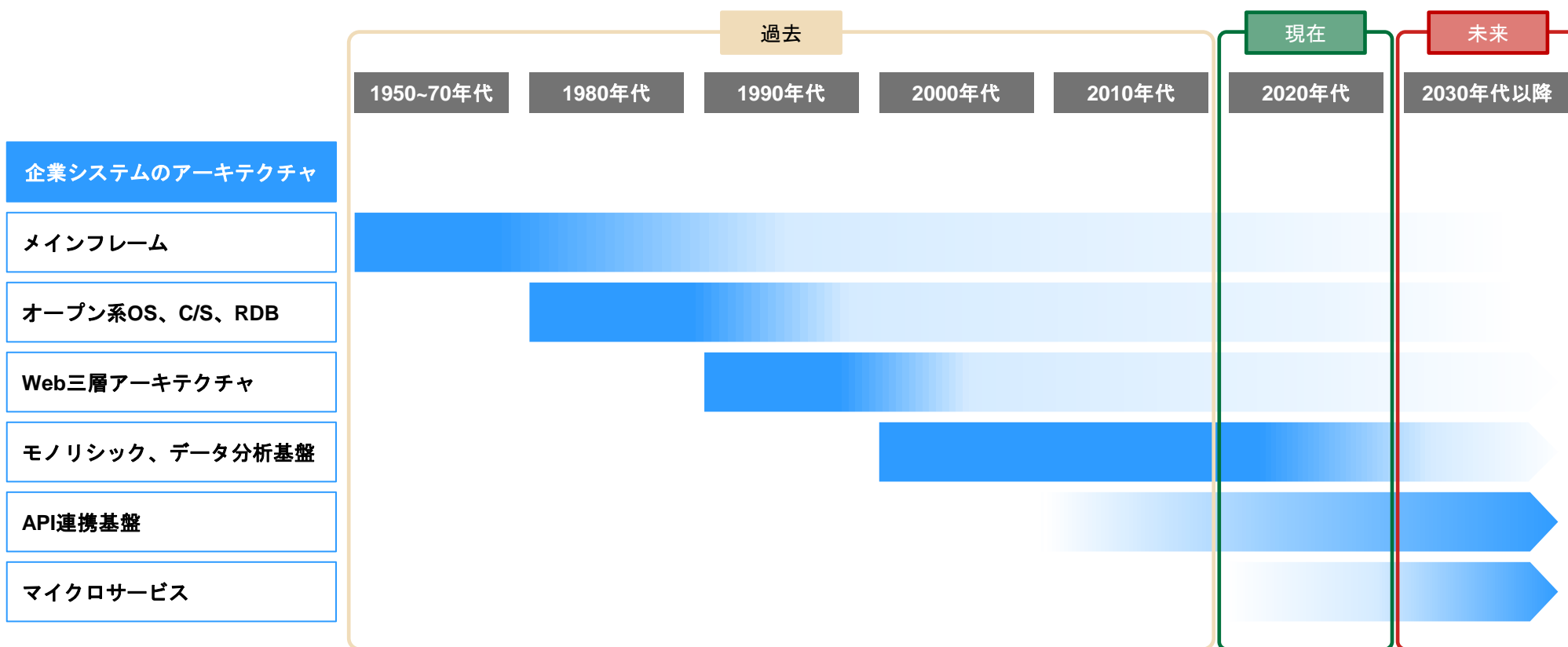
2025年8月28日

<はじめに>

柔軟性の高いアーキテクチャへの拡張

- 企業システムのアーキテクチャは、システムの柔軟性を高められる「API連携基盤」と「マイクロサービス」が主流となろう
- もっとも、社内のアプリケーションをモノリシックからマイクロサービスへ移行するためには工夫が必要と考える

企業システムのアーキテクチャの普及見通し



注1: 普及状況は、色が濃くなるに従って広範囲で普及している状態を示す

注2: 「OS」はOperating System、「C/S」はクライアントサーバ、「RDB」はRelational Database、「API」はApplication Programming Interfaceを指す © Nomura | STRICTLY PRIVATE AND CONFIDENTIAL

出所: 各種資料より野村證券作成

企業システムとデータ活用とは？

- 企業システムは企業が運用するITシステムで、複数のサーバで構成される。データは人とサーバやサーバ同士がやり取りする情報を指す
- データ活用とは、企業システムを動作して蓄積したデータを分析し、分析で得た知見を機能やサービスの更新・新規開発に活かす工程を指す

企業システム：企業が独自で運用するITシステム

- システムとは、複数のハードウェアやソフトウェアが連携して、ある特定の目的を達成するために設計された仕組みの全体像を指す
- システムを構成するハードウェアの一つがサーバ(コンピュータ)である
- サーバは、管理や予測を目的とした計算やデータ処理を行うためのハードウェアである

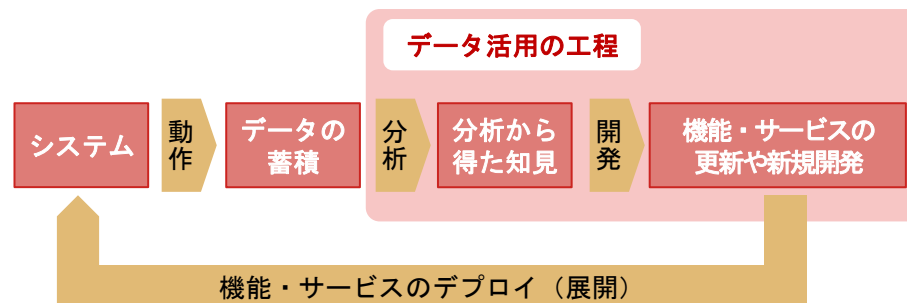
データ：リクエストとレスポンスという形でやり取りされる情報

- データとは、リクエストとレスポンスという形で、人とサーバやサーバ同士がやり取りする情報で、主に以下の3種類で構成される
 1. ユーザ(クライアント)がサーバへ入力する情報 (例：個人情報の入力)
 2. サーバが処理した結果として出力された情報 (例：レコメンドの生成)
 3. サーバの処理を指示するプログラムや命令 (例：計算の要求)
- リクエストとレスポンスは、クライアントとサーバ間でやり取りされるデータ通信の基本単位を指す
 - ✓ リクエストとは、クライアントがサーバに対して行う要求で、データや機能、サービスを求める操作を指す
 - ✓ レスポンスとは、リクエストに対するサーバからの応答で、要求されたデータや機能、サービスといった結果を出力する動作を指す

注：「機能」とは、システムが持つ個々の能力や動作を指す (例えば、スマートフォンの写真撮影機能)。
 「サービス」とは、機能を利用して提供される価値や体験を指す (例えば、写真共有サービス。撮影した写真をインターネットを通じて共有する)
 出所：各種資料より野村證券作成

データ活用：データから得た知見を機能・サービス開発に活かすこと

- データ活用とは、システムが動作した結果として出力されたデータを分析し、分析結果から得られた知見を、既存の機能やサービスの更新、新たな機能やサービスの開発に活かす作業である



出所：各種資料より野村證券作成

1. 企業システムのアーキテクチャの歴史



企業システムは科学的管理法の実践装置として始まった

- 企業システムは科学的管理法を実践するための装置として始まった
- 科学的管理法とは、客観的且つ定量的に生産現場のリソースや業務を管理することで業務効率を追求する経営手法である

企業システムでの科学的管理法の実践

- 科学的管理法は、生産現場のリソースや業務を、経験や勘に頼るのではなく、客観的且つ定量的に管理して「業務の効率化」を図る経営手法で、20世紀初頭にフレデリック・テイラーが提唱した

科学的管理法の具体例	企業システムの登場前の手段	科学的管理法に対応した企業システムの構成要素
労働時間の管理	紙やペン(インク)でタイムカードに打刻する	勤怠管理システム
現場作業の管理	紙やペンで描写したガントチャートで作業の進捗を管理する	タスク管理システム
作業や業務の標準化	紙製の業務マニュアルを作成する	業務アプリケーション ERPシステム
情報の記録や管理	紙製の文書に記録し、 ファイルした文書を書庫に保管する	DBMS(RDB、NoSQL)、DL(Apache Spark等)、 DWH(Snowflake、BigQuery等) DLH(Databricks等)、ETLツール
蓄積した情報の分析	保管した文書を分析する	BIツール(Tableau、Power BI、Looker等) AI分析ソフトウェア(AWS SageMaker、Azure Machine Learning等)

注1: 「ERP (Enterprise Resource Planning) システム」とは、企業の会計や人事、生産、販売等の基幹業務を統合的に管理し、効率化を図るためのシステム

注2: 「DBMS (Database Management System)」は、データを保存したり、検索、更新、削除するソフトウェア。「DL (Data Lake)」は、構造化データや半構造化データ、非構造化データ等の様々なデータをそのままの形式で格納したDB。「DWH(Data Warehouse)」は、複数のDBやDLからデータを集約し、分析に適したデータに加工・変換したDB。「DLH (Data Lake House)」は、DLとDWHの利点を組み合わせたDB。「ETL (Extract, Transform, Load) ツール」は、複数のDBやDLに格納したデータを収集し、分析に適した形式に加工・変換してDWHに格納するソフトウェア

出所: 各種資料より野村證券作成

企業システムは役割の拡大に合わせて進化を遂げてきた

- 企業のITシステムは、企業経営の根幹を支える基盤となっている
- IT技術の進歩と相まって、企業システムの役割が拡大した。それに合わせて、アーキテクチャも進化を遂げてきた

企業システムの進化史

	1950~70年代	1980年代	1990年代	2000年代	2010年代	2020年代
企業システムの役割	データによる業務管理、自動化	部門単位での導入と業務管理、自動化	グローバル単位での業務管理、自動化	ビッグデータの処理と分析	API連携で拡充した機能やサービスの運用	機動的な機能やサービスの開発の促進
アーキテクチャ	メインフレーム	オープン系、C/S、RDB	Web三層アーキテクチャ	モノリシックデータ分析基盤	API連携基盤	マイクロサービス
主要なIT技術	<ul style="list-style-type: none"> ・メインフレームの登場 	<ul style="list-style-type: none"> ・オープン系OS (UNIX、Windows) の登場 ・クライアントサーバ型アーキテクチャの登場 	<ul style="list-style-type: none"> ・インターネットの登場 (グローバルへの対象市場の拡大) ・ERPパッケージの登場 ・仮想化技術 (ハイパーバイザー) の登場 	<ul style="list-style-type: none"> ・クラウドサービスの登場 (SaaS、IaaS、PaaS) 	<ul style="list-style-type: none"> ・JSON APIの仕様の統一化 ・コンテナランタイムとコンテナイメージの標準化作業の完了 ・機械学習によるデータの分析能力の向上 	<ul style="list-style-type: none"> ・生成AIの登場 ・LLMとRAGの連携 ・AI Agentの登場と開発者向けプラットフォームの登場
主要なイベント	<ul style="list-style-type: none"> ・IBM System/360等のメインフレームの誕生 	<ul style="list-style-type: none"> ・NECや富士通等による国産オフコンの登場 ・オープンシステム化、マルチベンダー化したオフコンの登場 	<ul style="list-style-type: none"> ・SAPジャパン (1992) の設立、多言語・多通貨対応のERPの発売 ・国産ERPの発売 (OBIC7、GLOVIA、GRADIT、Bizj) ・VMware Workstation 1.0 (仮想化環境構築ソフトウェア) の発売 (1999) ・Salesforce (SaaS) の発売 (1999) 	<ul style="list-style-type: none"> ・AWS (IaaS) の発売 (2006) ・Google Cloud (IaaS) の発売 (2008) ・Google App Engine (PaaS) の発売 (2008) ・Amazon.com、Googleの台頭 (EC、デジマケ市場の創造) 	<ul style="list-style-type: none"> ・Microsoft Azure (IaaS) の発売 (2010) ・Microsoft 365 (SaaS) の発売 (2011) ・多層ニューラルネットワークの実現 (2012) ・Google Kubernetes Engine (GKE)、Amazon Elastic Container Service for Kubernetesの発表 (2018) 	<ul style="list-style-type: none"> ・RAGの論文発表 (2020年) ・GPT-3.5を搭載したChatGPTのリリース (2022年) ・AWS Bedrock Agentの発表 (2024年) ・Azure AI Foundryの発表 (2024年) ・Google Agentspaceの発表 (2024年)

注:「LLM」はLarge Language Model、「RAG」はRetrieval-Augmented Generationの略

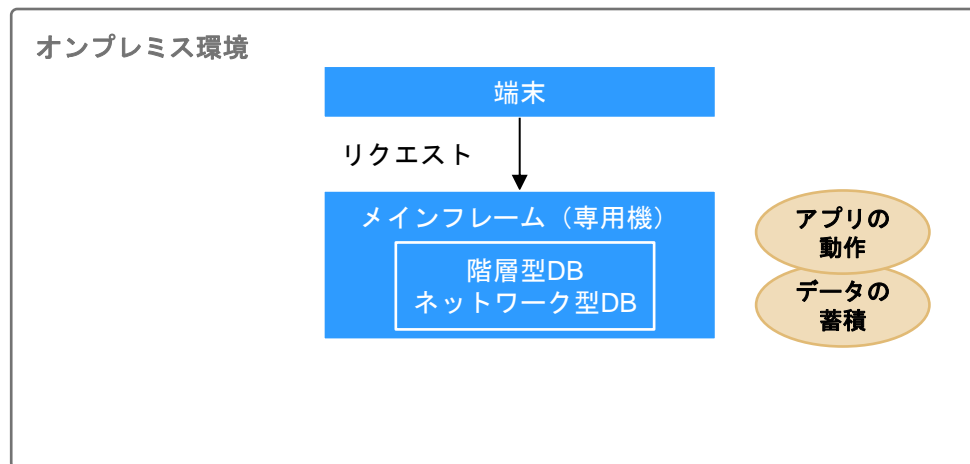
注2:RAGの論文とは、Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks"を指す

出所:各種資料より野村證券作成

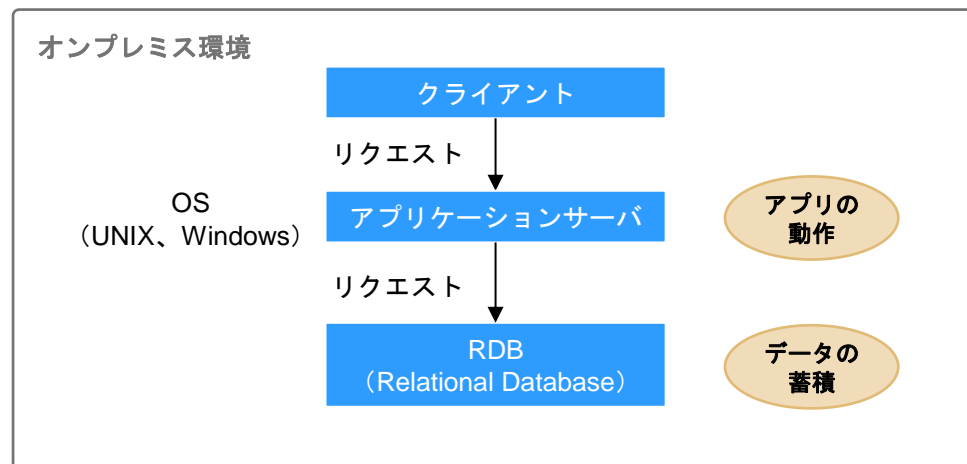
企業システムのアーキテクチャの変遷① 1950年代～1980年代

- 企業システムの歴史はメインフレームの登場から始まった。メインフレームは、特定の業務を遂行するための専用装置で、OSやアプリケーション、データベースが一体となったシステムであった
- 1980年代にオープン系OSやC/Sが登場し、企業システムの汎用化が進んだ。データベースではRDBが登場した

1950年代～1970年代：メインフレームの登場



1980年代：オープン系OS、C/S、RDBの登場



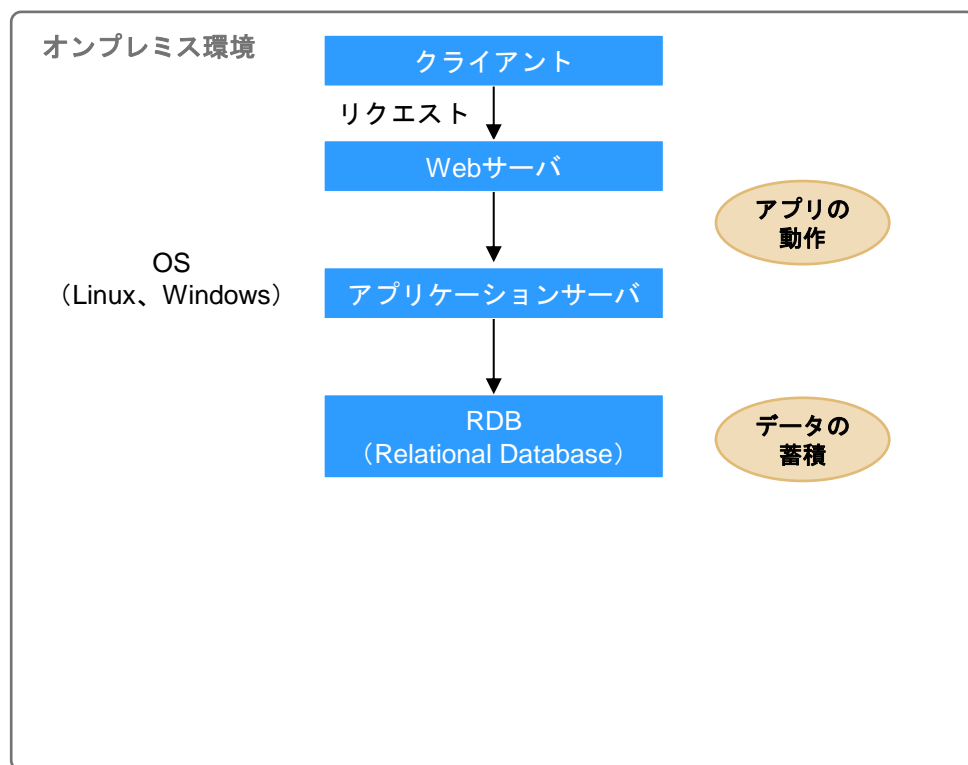
注：クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスがあるが、上図はリクエストのみを描写している。黒色の矢印はリクエストを指す
出所：各種資料より野村證券作成

注：クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスがあるが、上図はリクエストのみを描写している。黒色の矢印はリクエストを指す
出所：各種資料より野村證券作成

企業システムのアーキテクチャの変遷② 1990年代～2000年代

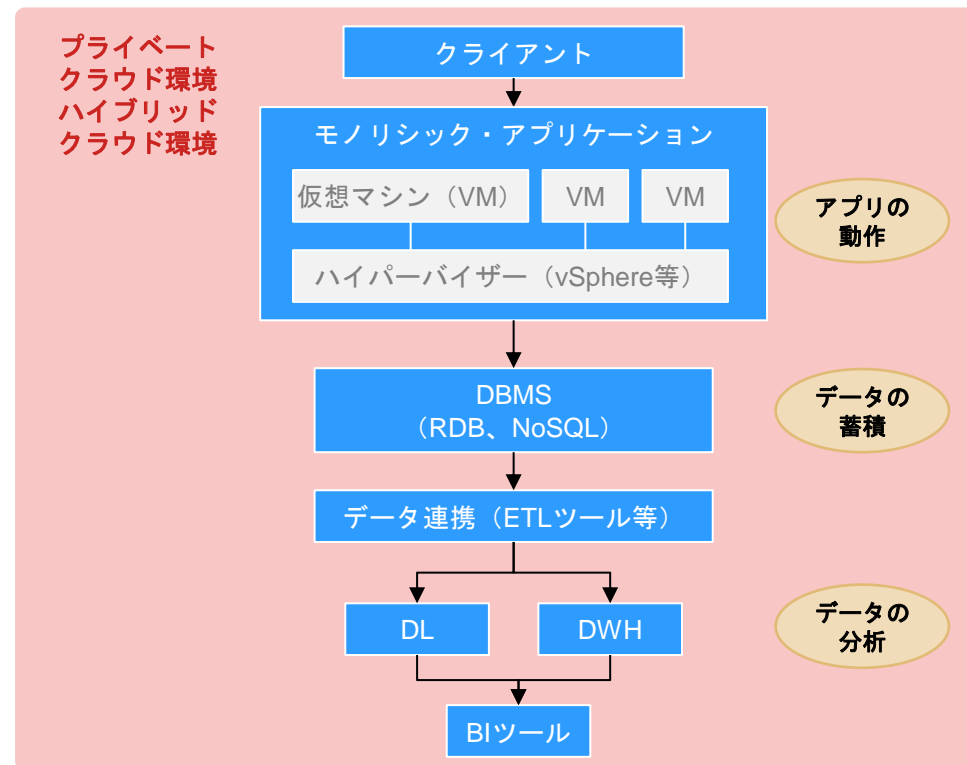
- 1990年代には、インターネット通信でシステムを運用する仕組みが普及。グローバルでの業務管理を目的に、ERPパッケージの導入が広まった
- 2000年代には、仮想化技術の応用したアプリケーションやデータベース、クラウドが登場し、ビッグデータを処理する基盤の整備が広まった。ECやデジタルマーケティング等のデータ活用ビジネスが登場したことでデータ活用への関心が高まり、データ分析基盤の整備も広まった

1990年代： Web三層アーキテクチャーの登場



注：クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスがあるが、上図はリクエストのみを描写している。黒色の矢印はリクエストを指す
出所：各種資料より野村證券作成

2000年代： モノリシック・アプリケーション、データ分析基盤の登場



注1：クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスがあるが、上図はリクエストのみを描写している。黒色の矢印はリクエストを指す

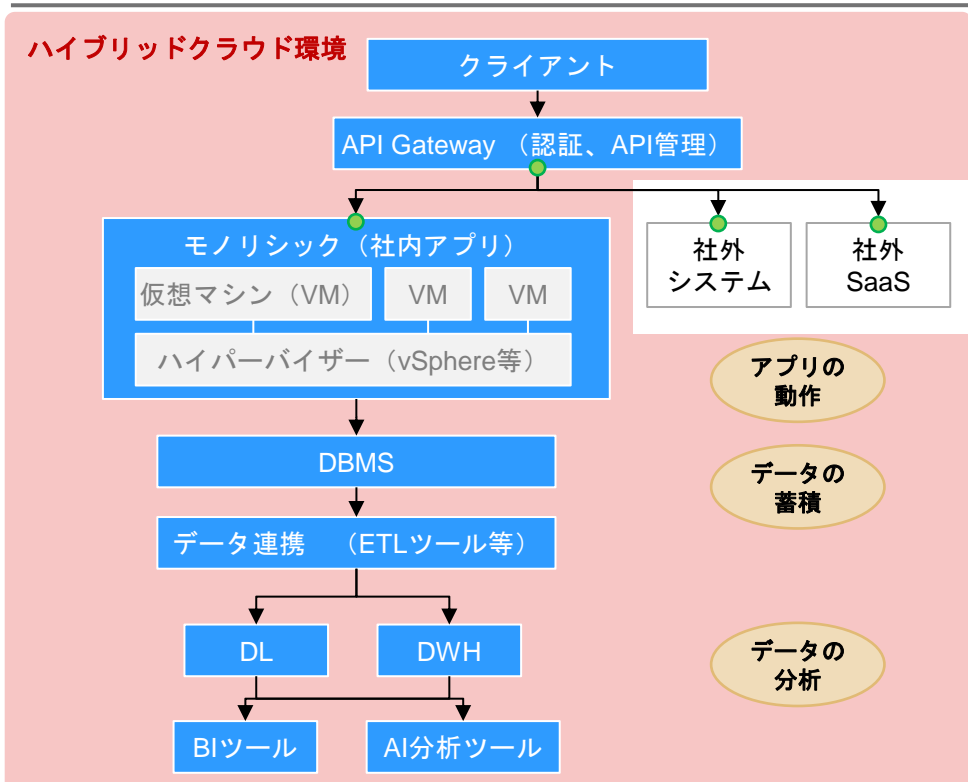
注2：「仮想マシン (VM)」は、1台のサーバのリソースをソフトウェアで分割し、あたかも別のサーバのように動作させる技術 (仮想化技術) を応用したリソースを指す。「ハイパーバイザー」は仮想マシンを制御するソフトウェアである

出所：各種資料より野村證券作成

企業システムのアーキテクチャの変遷③ 2010年代～2020年代

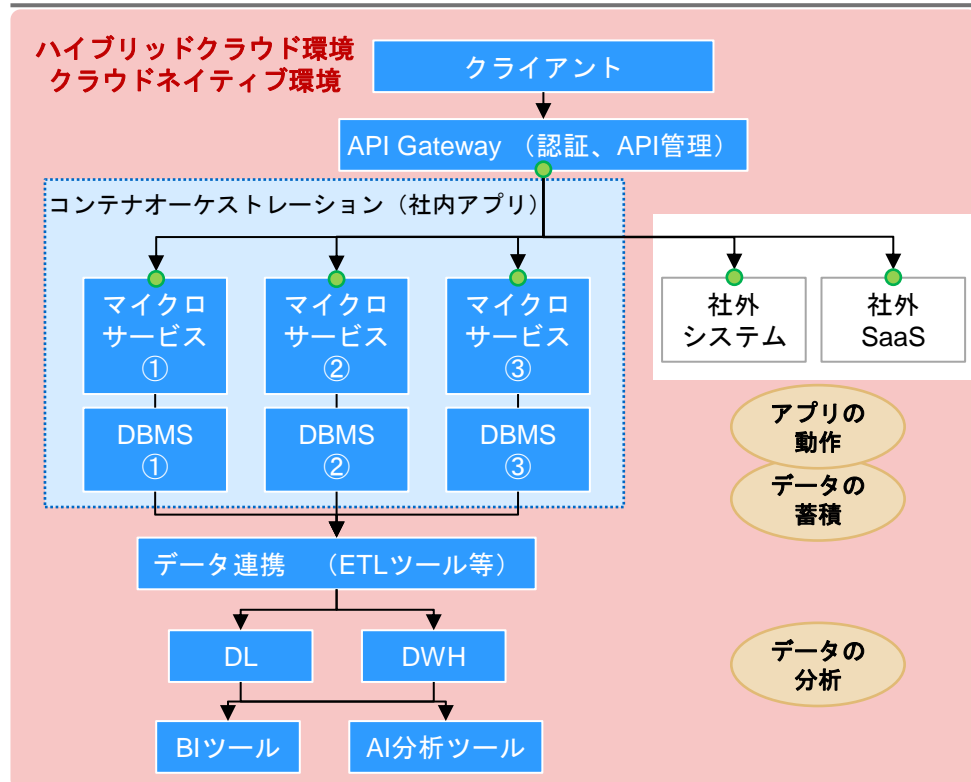
- 2010年代には、APIで社内外の機能やサービスを連携させることにより、事業ドメインを拡充したり、事業の付加価値を高める取組みが広まった。機械学習を中心としたAIによるデータ分析の高度化も試みられた
- 2020年代には社内のアプリケーションを改良する動きが見られた。アプリケーションを構成する全ての機能が一体となった密結合のモノリシックから、機能をサービス単位で細分化して、お互いを繋ぎ合わせた(疎結合の)マイクロサービスへ移行する企業が出てきた

2010年代： API連携基盤の登場



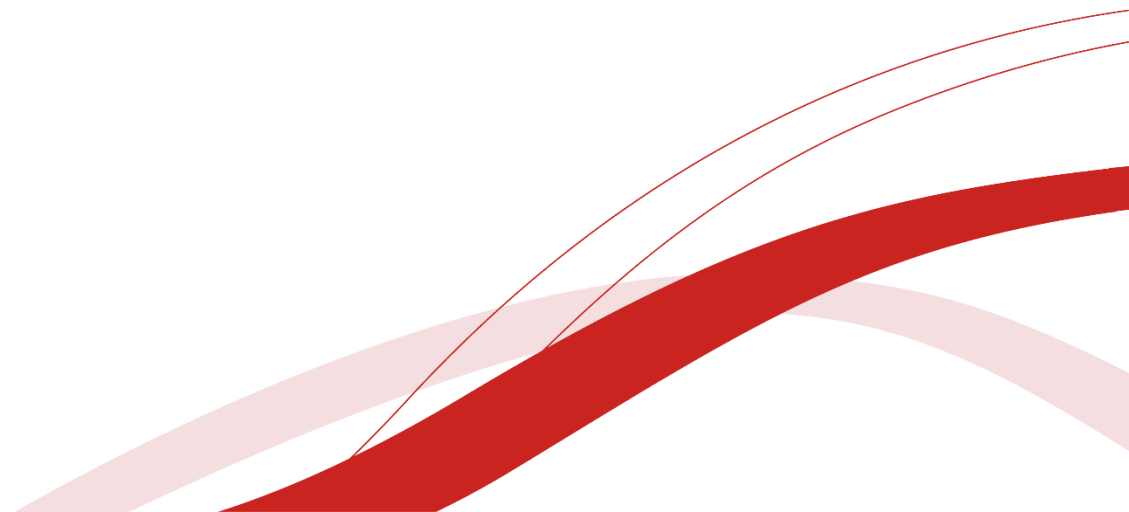
注1: クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスが、上図はリクエストのみを描写している。黒色の矢印はリクエストを指す
 注2: 「●」はソフトウェア同士がAPIで連携している状態を示す。API GatewayがDWHやBIツール等とAPI連携する場合もある
 出所: 各種資料より野村證券

2020年代： マイクロサービスの登場



注1: クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスが、上図はリクエストのみを描写している。黒色の矢印は人によるリクエストを指す
 注2: 「●」はソフトウェア同士がAPIで連携している状態を示す。API GatewayがDWHやBIツール等とAPI連携する場合もある
 出所: 各種資料より野村證券

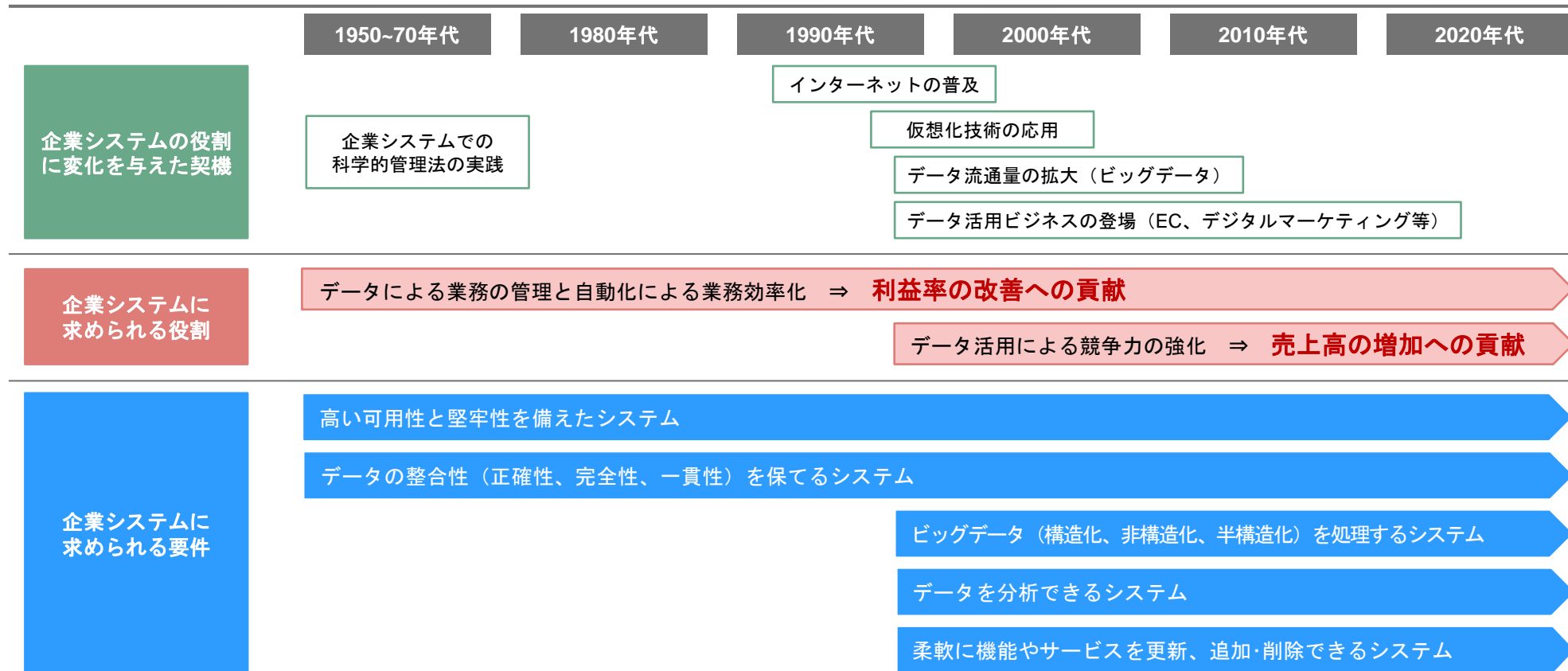
2. 企業システムの歴史観



2000年代を境に企業システムに新たな役割が加わった

- 企業は業務の管理や業務の効率化を追求すべく、企業システムを改良してきた
- 2000年代にインターネットや仮想化技術、データ活用ビジネスが登場したことを受け、企業はシステムにデータ活用による競争力の強化をサポートする役割も求めるようになった。企業システムに求める要件も増えた

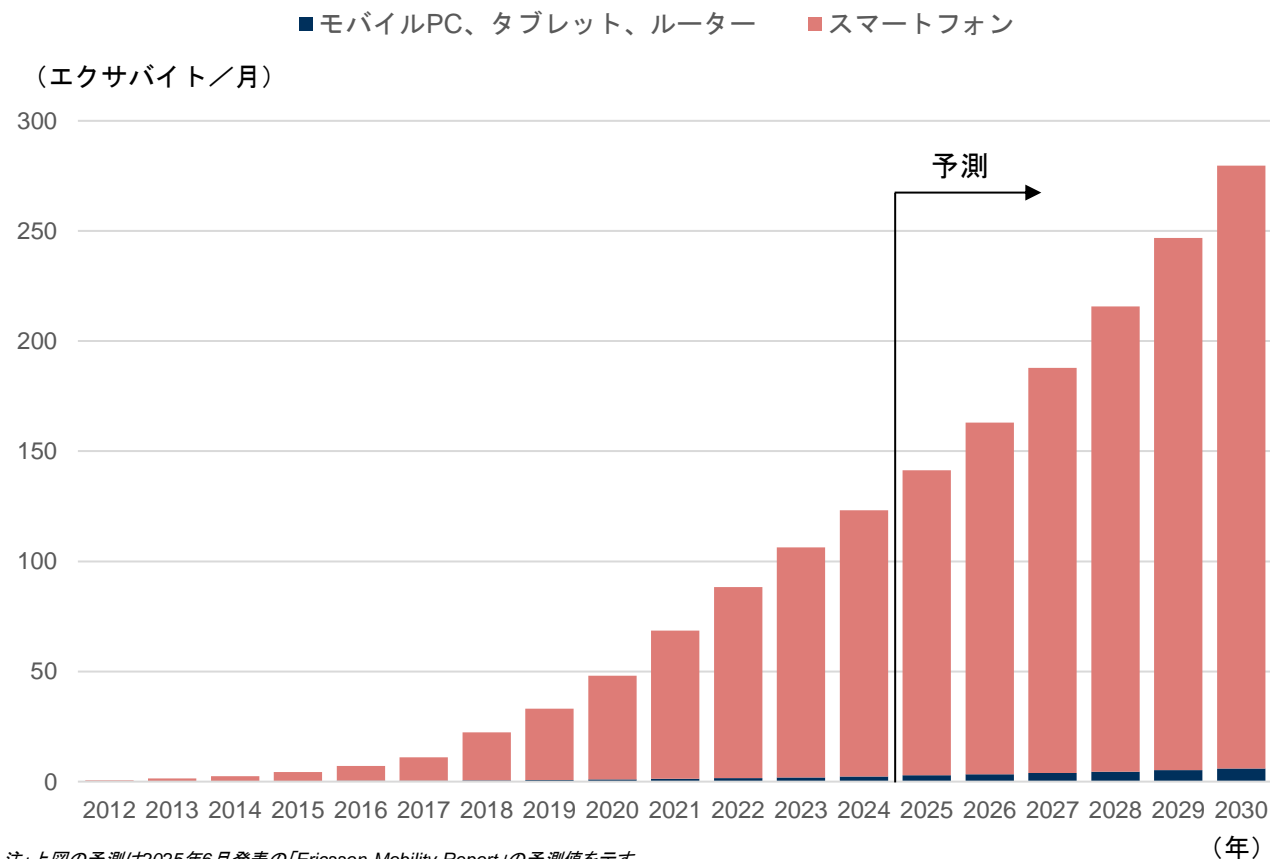
企業システムに求められる役割の変化



データ流通量の拡大

- Ericssonによると、スマートフォンの普及を背景に、2025年のデータ流通量は約140エクサバイト／月、2030年は約280エクサバイト／月への拡大が予想される
- データ流通量が拡大することで、機能やサービスの開発に資する有効なデータも増えよう

世界のモバイルデータの流通量の予測(デバイス別)

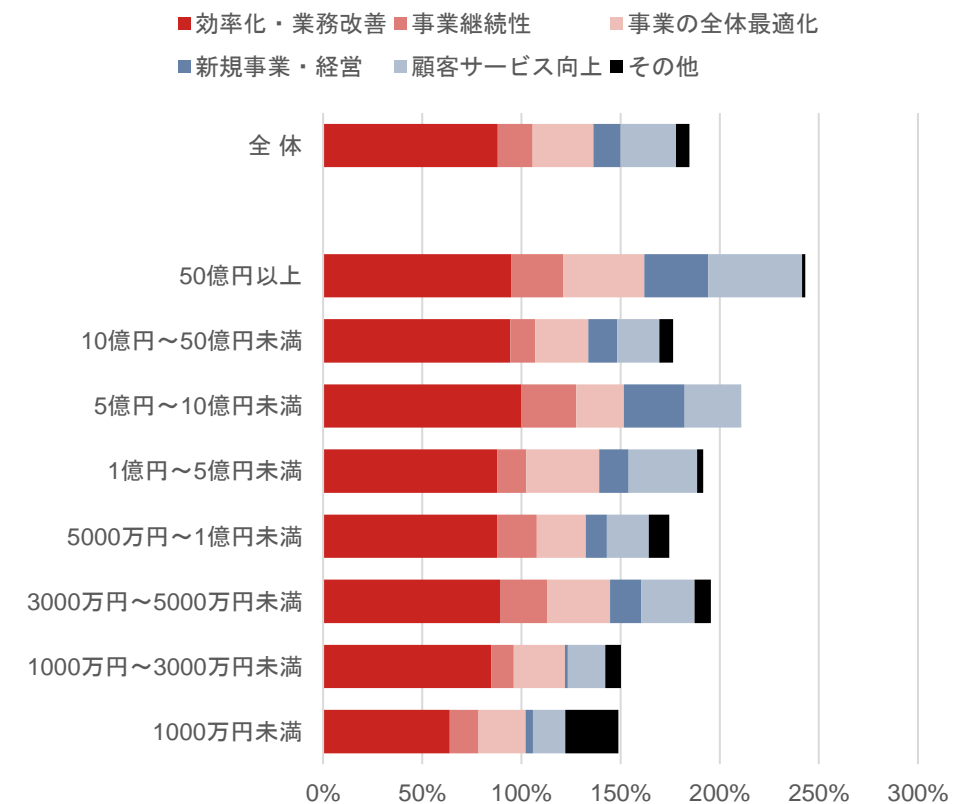
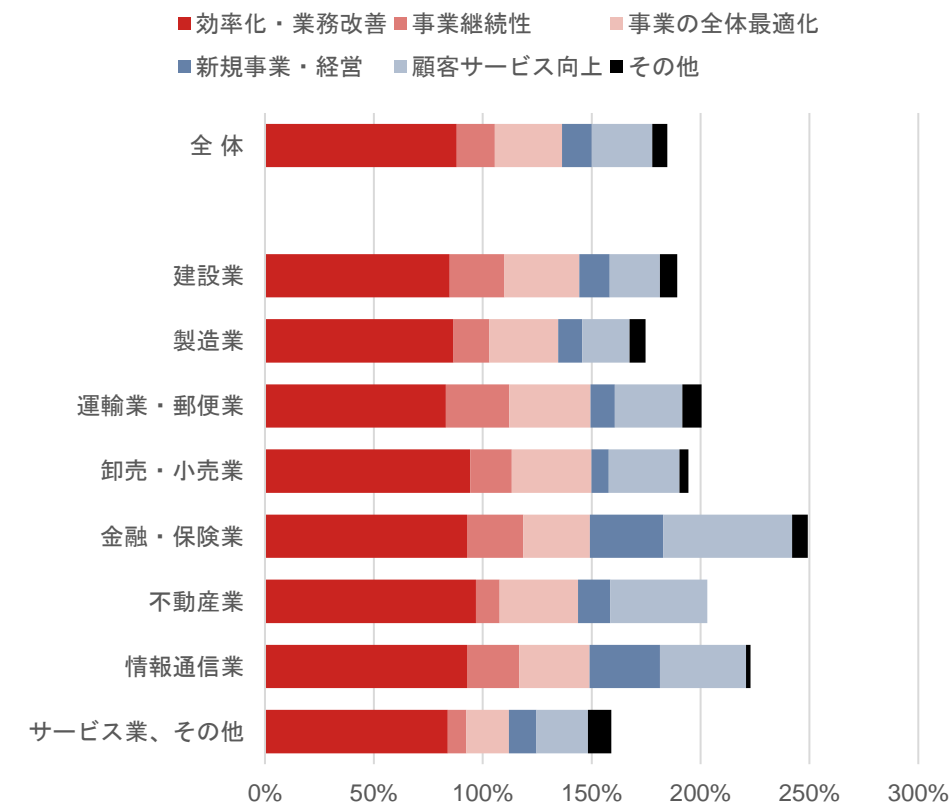


企業によるデータ活用の現状

- 企業のデータ活用目的は、業務の管理や業務の効率化が多いが、既存事業の強化や新規事業開発も存在する
- ITシステムによる業務の自動化が進む業種(情報通信業、金融・保険業)では既存事業の強化や新規事業開発にデータを活用する企業が多い
- 大企業には、既存事業の強化や新規事業開発にデータを活用する企業が多い

業種別 デジタルデータの収集・解析目的(複数回答可)

資本金別 デジタルデータの収集・解析目的(複数回答可)



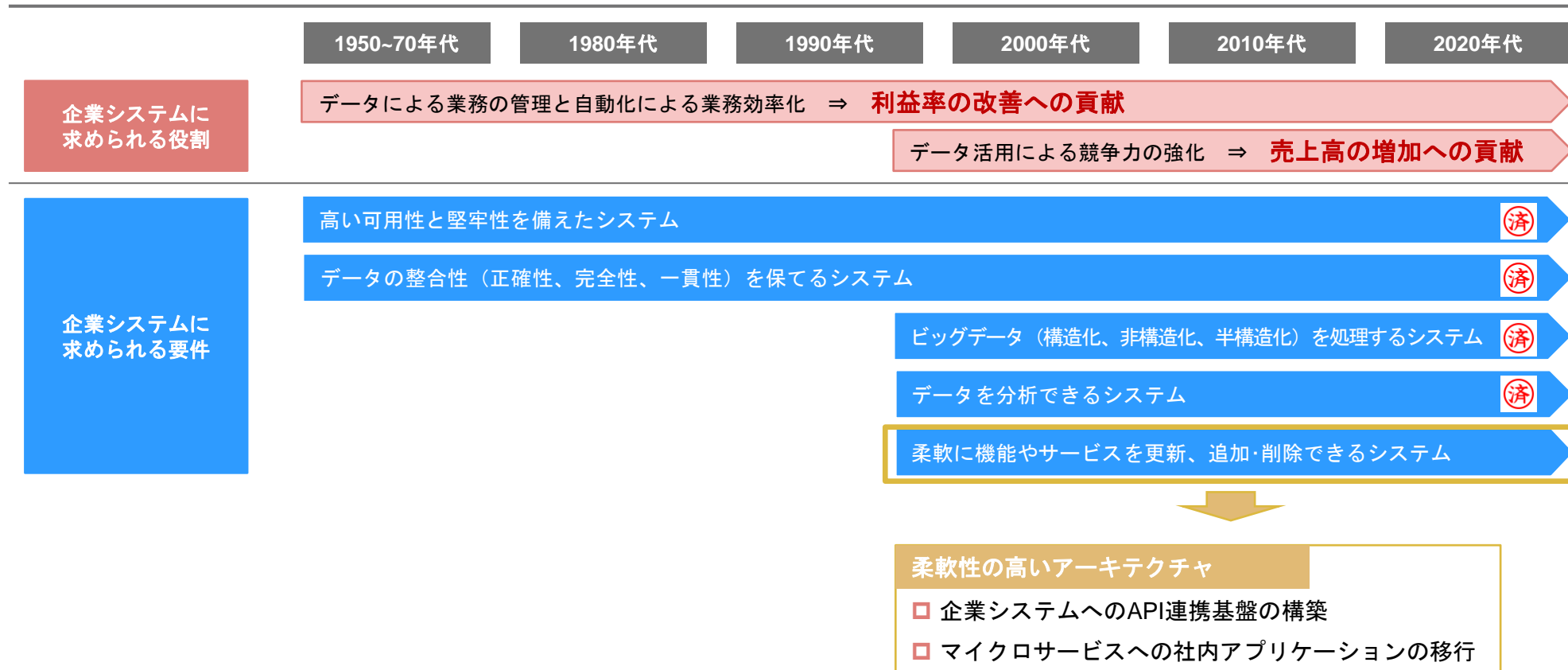
出所:総務省「令和6年通信利用動向調査」より野村證券作成

出所:総務省「令和6年通信利用動向調査」より野村證券作成

企業システムには『柔軟性』が求められている

- データ流通量の拡大を受けて、企業は既にビッグデータの処理やデータを分析できるシステムを整えていると見られる
- 今後、企業は、データ活用による競争力の強化に向けて、データを分析して得た知見をもとに、機動的に機能やサービスを更新、追加・削除できる柔軟性の高いシステムを追求するであろう

企業システムに求められる要件の変化



柔軟性の高いアーキテクチャへの拡張

- 柔軟性の高いシステムとは、柔軟にアプリケーションを変更できるシステムを指す
- 柔軟性の高いシステムを実現するためには、企業システムにAPI連携基盤を構築することや社内のアプリケーションをモノリシックからマイクロサービスへ移行することが有効である

柔軟性の高い企業システムへ拡張する手段

柔軟性の高い企業システム

データを分析して得た知見をもとに、柔軟にアプリケーションを変更できるシステム

1

社内外の様々な機能やサービスをAPIで連携させることで、事業ドメインを拡充できる、事業の付加価値を強化できるシステム

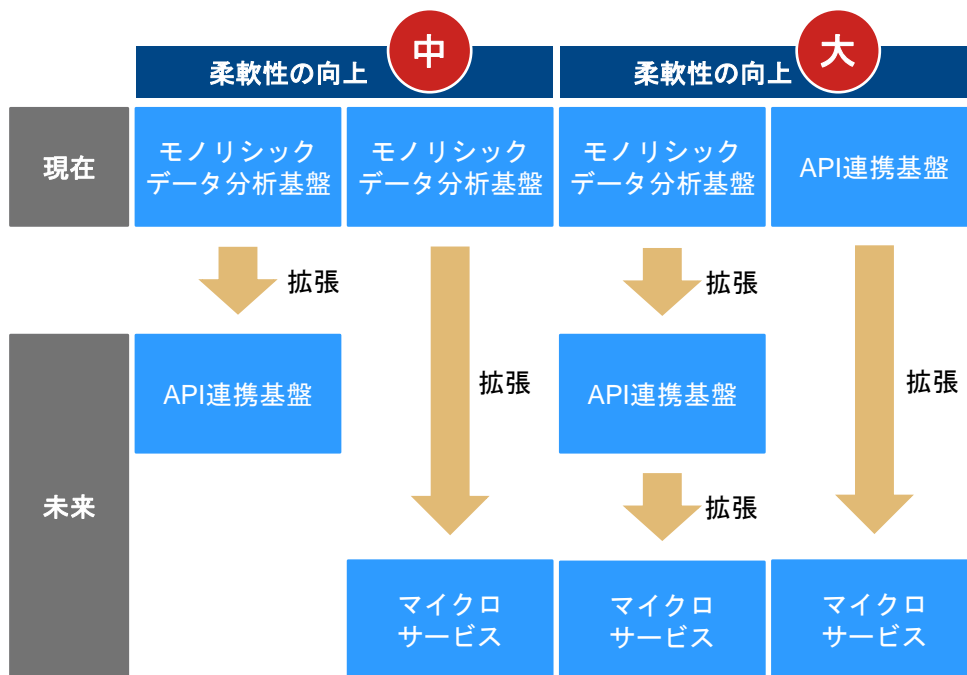
API連携基盤の構築

2

社内のアプリケーションを構成する機能毎に更新できる、新たな機能を追加や削減も可能なシステム

モノリシックからマイクロサービスへの移行

アーキテクチャの現在と未来のパターン



API連携基盤とマイクロサービスは共に、クラウド環境で運用することが望ましい。クラウド環境は、オンプレミス環境と比べてスケーラビリティが高いため、API連携する機能やサービスを追加、削除し易いためである

3. 柔軟性の高いアーキテクチャの具体像

～企業システムへのAPI連携基盤の構築、マイクロサービスへの社内アプリケーションの移行～



API連携基盤の構築に必須のAPI Gateway

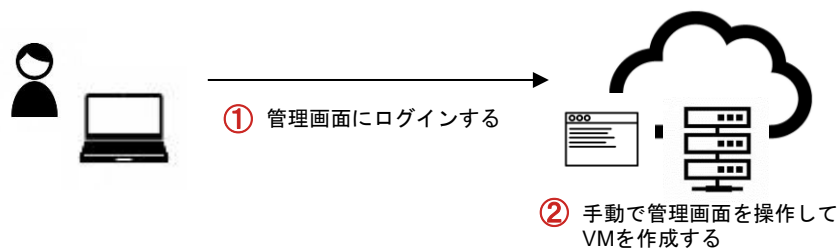
- APIにより社内外の様々な機能やサービスを連携させることが可能である
- API Gatewayを設置することで、連携する機能やサービスを管理し易くなり、機動的に機能やサービスの更新、追加・削除も容易になる

APIの概要

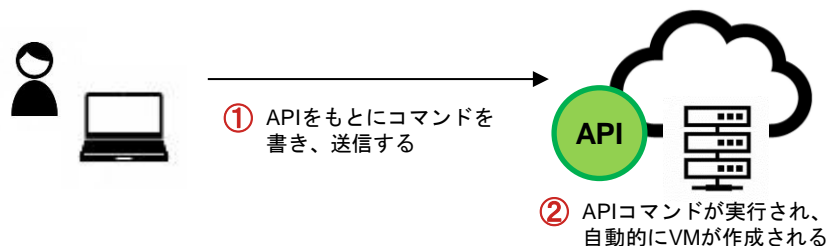
- APIとは、あるプログラムが持つ機能やサービスを、外部にある他のプログラムから呼び出して利用するためのコマンドや関数、データ形式等を定めた規約である

<例> 仮想マシン(VM)を作成する

【手動で操作】 クラウド事業者の管理画面を操作して手動で仮想マシンを作成する



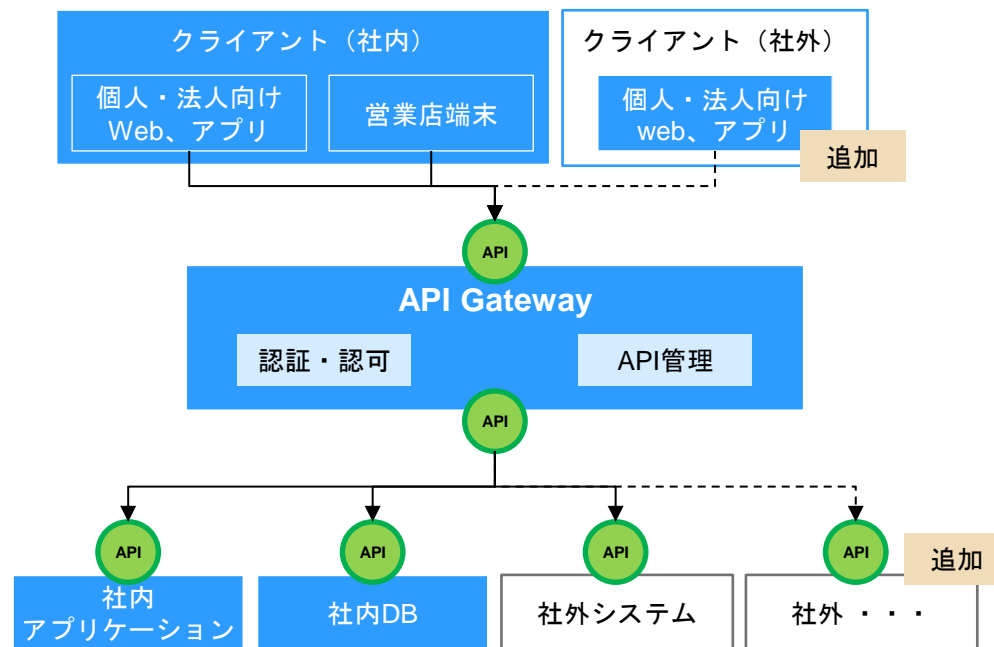
【API連携】 クラウド事業者が用意したAPIをもとにプログラムを作成して、そのプログラムで仮想マシンを作成する



出所: 各種資料より野村證券作成

API連携基盤の構造 ~API Gatewayは仲介役~

- APIで社内外の様々な機能やサービスを連携させ、管理するには手間がかかる
- API Gatewayは、フロントエンド(クライアント)とバックエンドの仲介役を担い、APIを一元的に管理・制御する

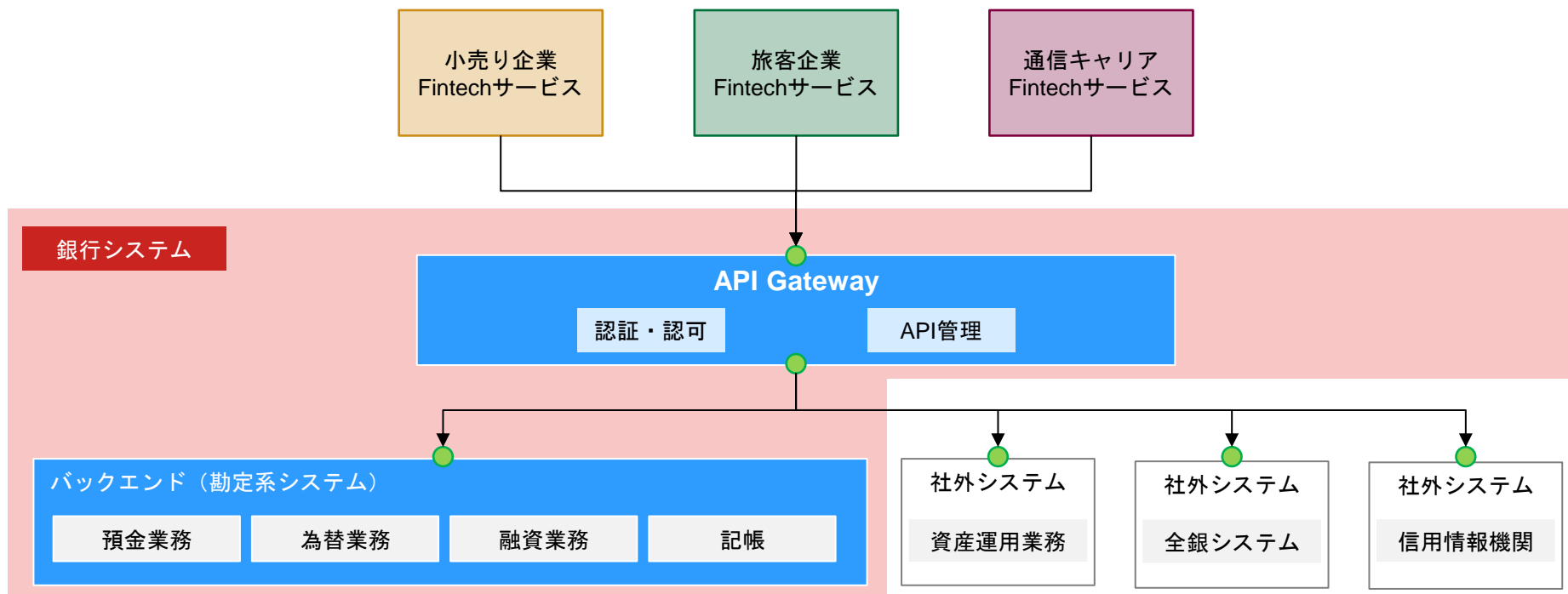


出所: 各種資料より野村證券作成

API連携基盤の応用例 ～BaaS(Banking as a Service)～

- BaaS(Banking as a Service)は、銀行の機能やサービスを、主に銀行免許を持たない小売りや旅客等の企業のサービスに組み込んで利用できるようにする仕組みを指す
- 銀行はAPI Gatewayを設置して、フロントエンドのFintechサービスとバックエンドの勘定系システムを繋ぎ、APIを一元的に管理・制御する

BaaS(Banking as a Service)のアーキテクチャ例



注1: クライアントとサーバ、サーバ同士の通信にはリクエストとレスポンスが、上図はリクエストのみを描写している

注2: 「●」はソフトウェア同士がAPIで連携している状態を示す

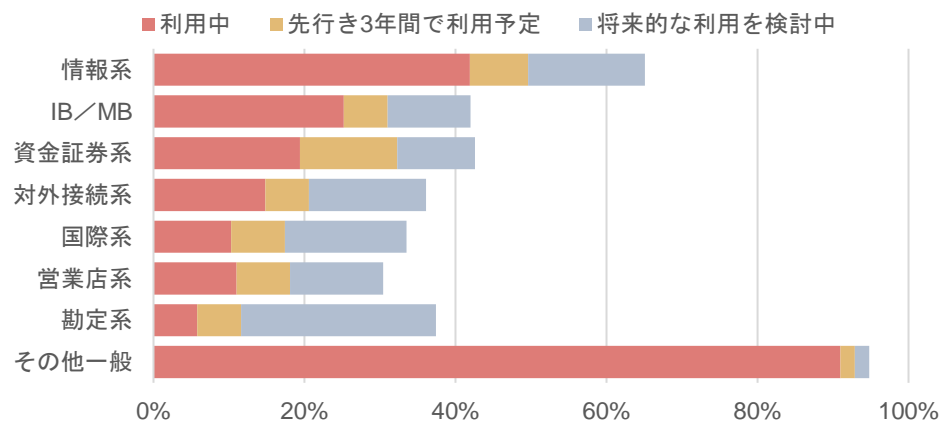
注3: 上図では銀行のシステムのうち、DBMS以下を除いて描写している

出所: 各種資料より野村證券作成

Slerに求められる開発力

- 多くの企業はハイブリッドクラウドを採用している。データ活用に積極的な金融機関でさえ、業務領域毎にオンプレミスとクラウドとを併用している
- 企業のシステム開発を支援してきたSlerには、API仕様や認証方式に対応していないオンプレミス環境のシステムとクラウド環境のシステムを繋ぐ開発力が求められる。また、顧客企業のシステムに適したAPI Gatewayの開発力も求められる

金融機関の業務領域別システムでのクラウドの利用状況



情報系	顧客管理や収益管理、リスク管理業務や融資審査業務等を含む機密性に配慮する必要があるシステム
IB/MB	インターネット・モバイルバンキングのシステム
資金証券系	資金、為替、債券、株式、デリバティブ等の資金・証券業務に関する取引や事務処理を支援するシステム
対外接続系	金融機関と外部の金融ネットワーク及び顧客ネットワークなどの接続を受け持つシステム 例えば、CAFISやCARDNETなどの決済センターや個人信用情報機関 (JICC/CIC/KSC) 等の対外センターと接続する
国際系	自社の海外支店の業務処理システムや海外の金融機関との決済のために使用されるシステム
営業店系	金融機関の窓口で利用される取引端末やATM などを含めたシステム
勘定系	入出金や資金の決済、口座や融資の残高管理、利息計算などの勘定処理を行なうシステム
その他一般	顧客情報や決済業務に直接関係しない社内業務システム。電子メールシステム、ファイルサーバシステム、Webサーバ等

注：調査先の業態は都市銀行、ネット銀行、地方銀行、信用金庫、信託銀行、信用組合の156金融機関
出所：日本銀行「金融機関におけるクラウドサービスの利用状況と利用上の課題について」より野村証券作成

Slerに求められる開発力

APIによるオンプレミス環境のシステムの連携

- オンプレミス環境への安全な接続経路の確保が必要である。専用回線やVPNで繋ぐことが一般的である
- APIの仕様や認証方式に対応していないオンプレミス環境のシステムには、API Adaptorを設置する必要がある

顧客企業のシステムに適したAPI Gatewayの開発

- クラウドサービスプロバイダーが提供するAPI Gatewayに、顧客企業が求めるOAuth等の認証・認可の機能を実装する必要がある
- 顧客企業が求める認証・認可の機能の開発には、システムの全体像や業務フローの理解が必要になる

注1:「VPN」とは、インターネット上に仮想的な専用の通信経路を作り、セキュリティを確保しながら安全にデータを送受信する技術を指す

注2:「API Adaptor」とは、APIを通じて、異なるシステム間の連携を可能にするソフトウェアコンポーネントを指す

注3:「OAuth」とはユーザーがパスワードを共有することなく、あるシステムが他のシステムにアクセスを許可するためのプロトコルを指す

出所：各種資料より野村証券作成

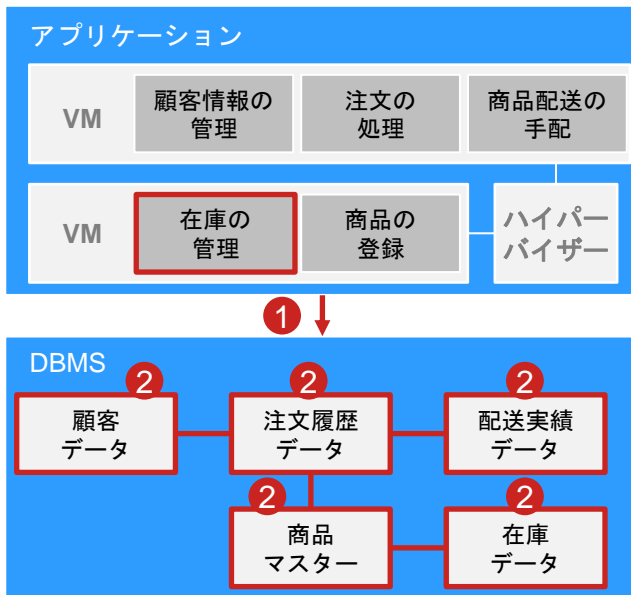
モノリシックとマイクロサービスの構造面の違い

- モノリシックは一枚岩を意味し、全ての機能が一体となってアプリケーションを成す構造を指す。其々の機能が密接に結合しており、密結合の構造と呼ばれる。そのため、1つの機能の更新が、アプリケーション全体に及ぼす影響は大きくなってしまふ
- マイクロサービスは、機能をサービス単位で細分化して、サービス同士が連携しながら機能を動かす構造である。其々の機能の依存関係が弱く、独立して動作するため、疎結合の構造と呼ばれる。そのため、1つの機能を更新する場合、全体に及ぼす影響を抑えることができる

モノリシック・アプリケーションのアーキテクチャ

- モノリシック・アーキテクチャとは、全ての機能が一体となって大きなアプリケーションを成す構造である
- 1つの機能が動作すると、全てのデータが即座にアップデートされる

<例> ECのアプリケーション

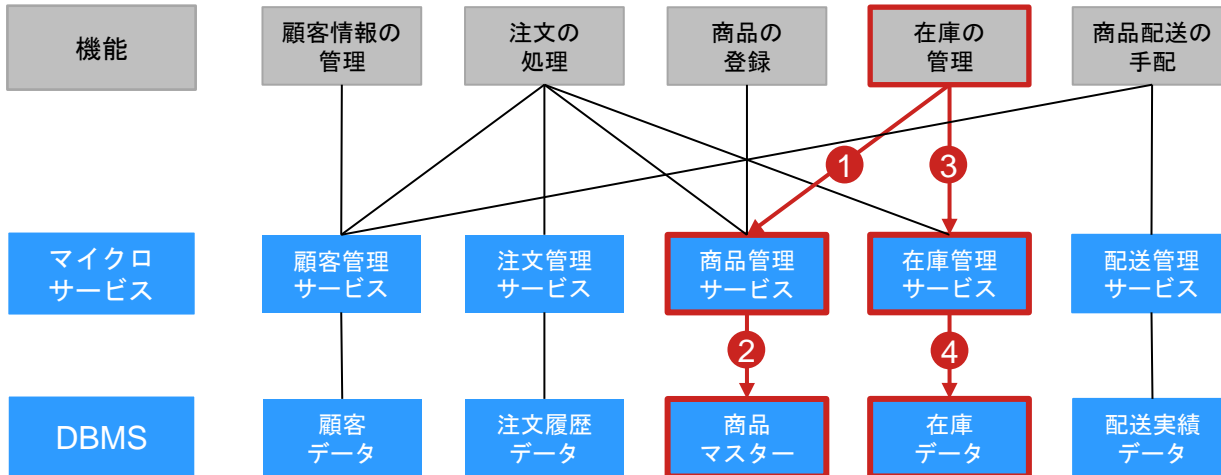


出所: 各種資料より野村證券作成

マイクロサービスのアーキテクチャ

- マイクロサービスのアーキテクチャは、アプリケーションを構成する機能と独立して動作する小さなサービスを連携させた構造である
- 1つの機能を動作すると、それに関連するサービスが順番にデータをアップデートしていく

<例> ECのアプリケーション



出所: 各種資料より野村證券作成

モノリシックとマイクロサービスの特性の比較

- 社内のアプリケーションをマイクロサービスへ移行することで、アプリケーション全体ではなく、それを構成する機能毎に更新できたり、新たな機能を追加・削除できるようになる。すなわち、機能の更新、追加・削減の機動性を高められる
- ただし、マイクロサービスが確保できるデータの整合性は結果整合性である。モノリシックからマイクロサービスへ移行する場合、強整合性から結果整合性へのスペックダウンを許容する必要がある

モノリシックとマイクロサービスの特性比較表

		モノリシック	マイクロサービス
構造		単純	複雑
定義		すべての機能を一つの大きなアプリケーションとした構造	機能と独立して動作するサービスが連携した構造
機能との結合関係		密結合	疎結合
柔軟性		低	高
機能の更新		単一のプロジェクトとしての開発になる	サービス毎に独立した開発ができる
機能の追加		機能追加の影響が大きいため全体的なテストが必要	新機能に関連するサービスのみ限定したテストを実施できる
機能のデプロイ(展開)		アプリケーション全体でデプロイ	サービス単位でデプロイ
機能のスケーリング		アプリケーション全体でまとめてスケール	必要なサービスに限定してスケール
技術の選択範囲		一つの技術スタックに限定される	サービス毎に異なる技術を選択できる
保証するデータの整合性		強整合性	結果整合性

注:「スケーリング」とは、システムやアプリケーションの処理能力や容量を、需要の変化に合わせて調整することを指す
出所:各種資料より野村證券作成

マイクロサービスはデータの結果整合性までしか確保できない

- データの整合性とは、データの正確性や完全性、一貫性が保たれている状態を指す。マイクロサービスでは機能の動作している間は一貫性を保つことができない
- マイクロサービスでは、サービス毎にデータベースが設置されており、機能の動作に合わせて順次データが更新される。そのため、データの整合性は結果整合性までしか確保できない(強整合性は確保することは難しい)

データの整合性とは ～強整合性と結果整合性に分かれる～

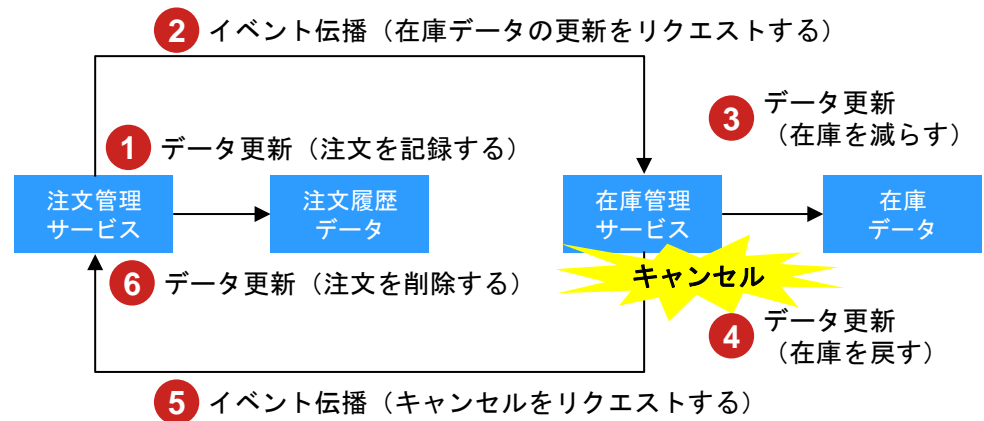
- データの整合性とは、データに誤りや不整合がなく(正確性)、必要なデータがすべて揃っており(完全性)、異なるデータベース間でデータが矛盾なく連携している(一貫性)状態を指す
- データの整合性は、一貫性が確保されるレベルによって、強整合性と結果整合性に分かれる

特性	強整合性	結果整合性
DBへのデータの反映	即座に全てのDBに反映される	最終的には全てのDBに反映される
データの一貫性	常に確保される	いつかは確保される
適合する場面	リアルタイムでの処理が必要なデータを扱う場面(決済や送金等)	リアルタイムでの処理が不要なデータを扱う場面(SNSのコメント等)

データベース間でデータを同期する流れ

<例> 注文を処理する途中でキャンセルされた場合

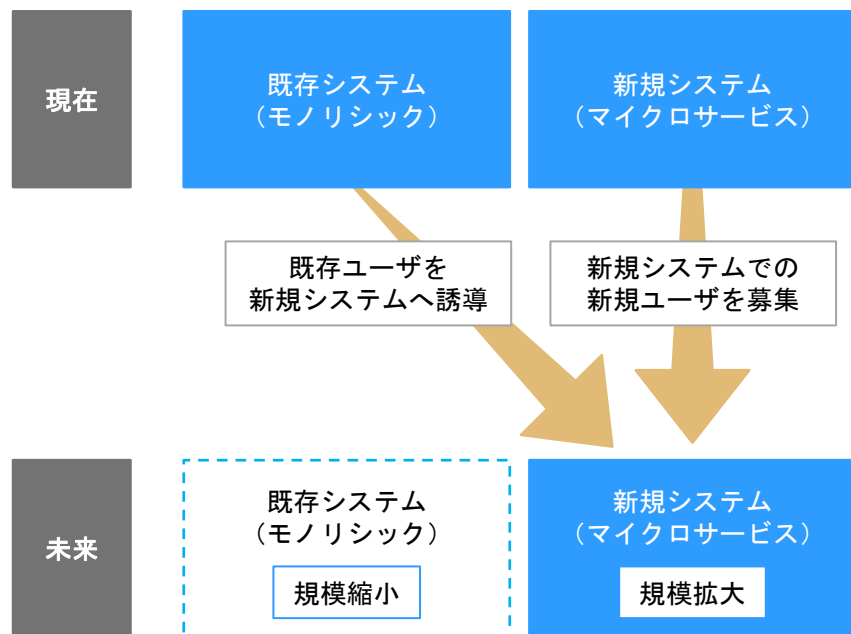
- 「注文処理」機能を実行する場合、「注文管理サービス」と「在庫管理サービス」を通じて、2つのデータを順番に更新する。具体的には、注文が確定した段階で、「注文履歴」の行を1つ追加し、「在庫データ」から在庫を1つ減らす
- データを更新した後に注文のキャンセルがあった場合、在庫データの在庫を戻したのちに、注文履歴の行を削除する作業が必要になる。在庫データを戻さなければ、在庫が減ったままの状態になり、データの一貫性が保たれない状態となる



マイクロサービスへの移行戦略 ～エンドユーザをマイクロサービスへ誘導する～

- 既存システムのアプリケーションをモノリシックからマイクロサービスへ移行するのではなく、エンドユーザにデータの結果整合性を許容してもらったうえで、エンドユーザをマイクロサービスを実装した新規システムに誘導し、将来は新規システムをメインとする戦略が検討されている
- 三菱UFJ銀行は、オンプレミスの既存の勘定系システムとマイクロサービスで構築した新たな勘定系システムの両方を運用しながら、新規顧客と既存顧客を新たな勘定系システムへ誘導すると見られる。将来は、(個人向けでは)新たな勘定系システムがメインとなろう

マイクロサービスへの移行戦略

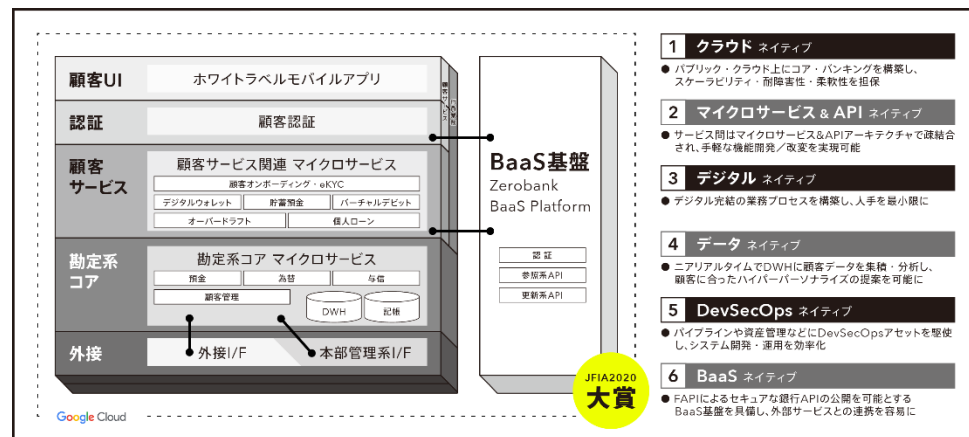


<留意点> 新規システムのエンドユーザは、データの結果整合性を許容する必要がある

出所: 各種資料より野村證券作成

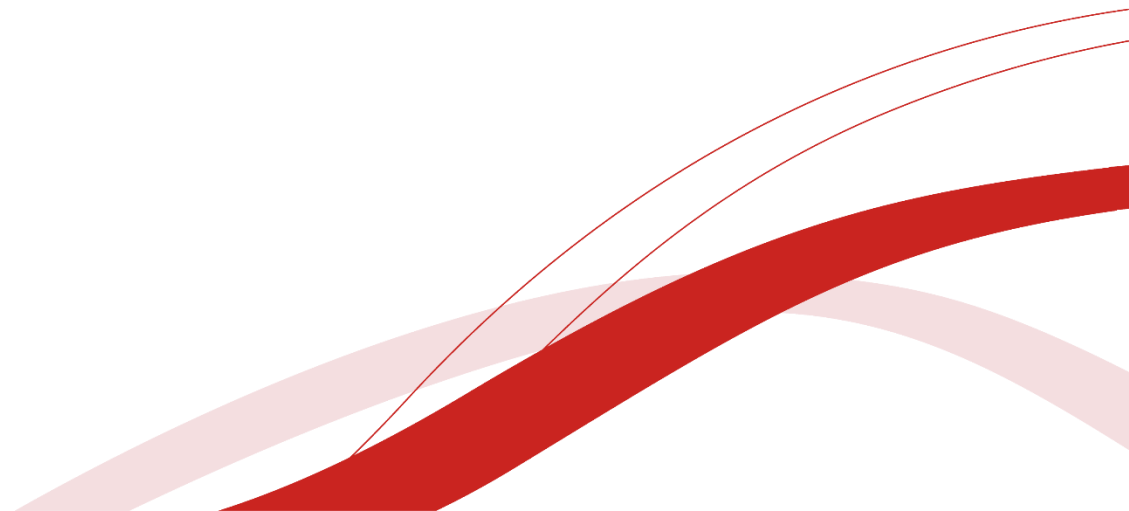
三菱UFJ銀行が導入するフルクラウド型銀行システムの概略図

- 2025年5月にみんなの銀行は自社開発のフルクラウド型銀行システムを三菱UFJ銀行へ提供することを発表した
- フルクラウド型銀行システムのアプリケーションはマイクロサービスを採用している



出所: みんなの銀行のプレスリリースより野村證券抜粋

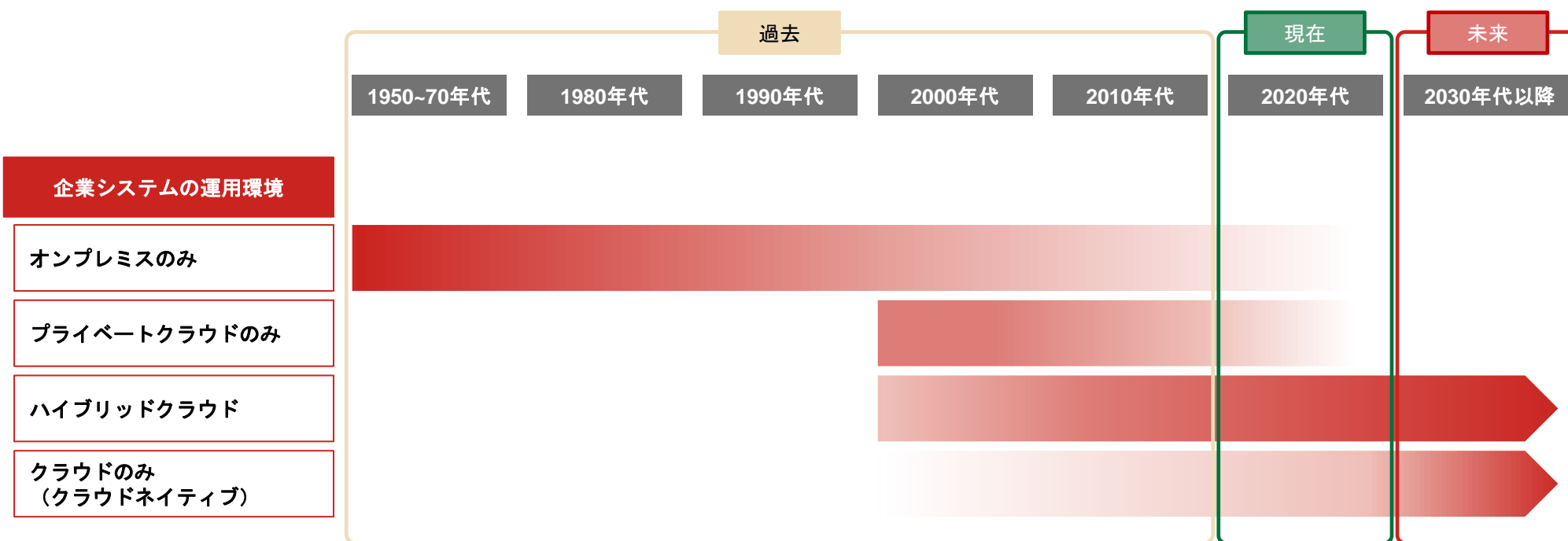
Appendix



企業システムの運用環境の現在と未来

- 2000年代以降、日本企業はシステムの運用環境をオンプレミスからクラウドへ移行してきた。ただし、機微情報を扱うアプリケーションやデータベース等は、引き続き、オンプレミスやプライベートクラウドでの運用を希望する企業が多く、現在はハイブリッドクラウドが主流である
- クラウドサービスの堅牢性の向上やAPI連携し易さから、クラウドネイティブの運用環境へ移行する企業が増えると考え

企業システムの運用環境の普及見通し

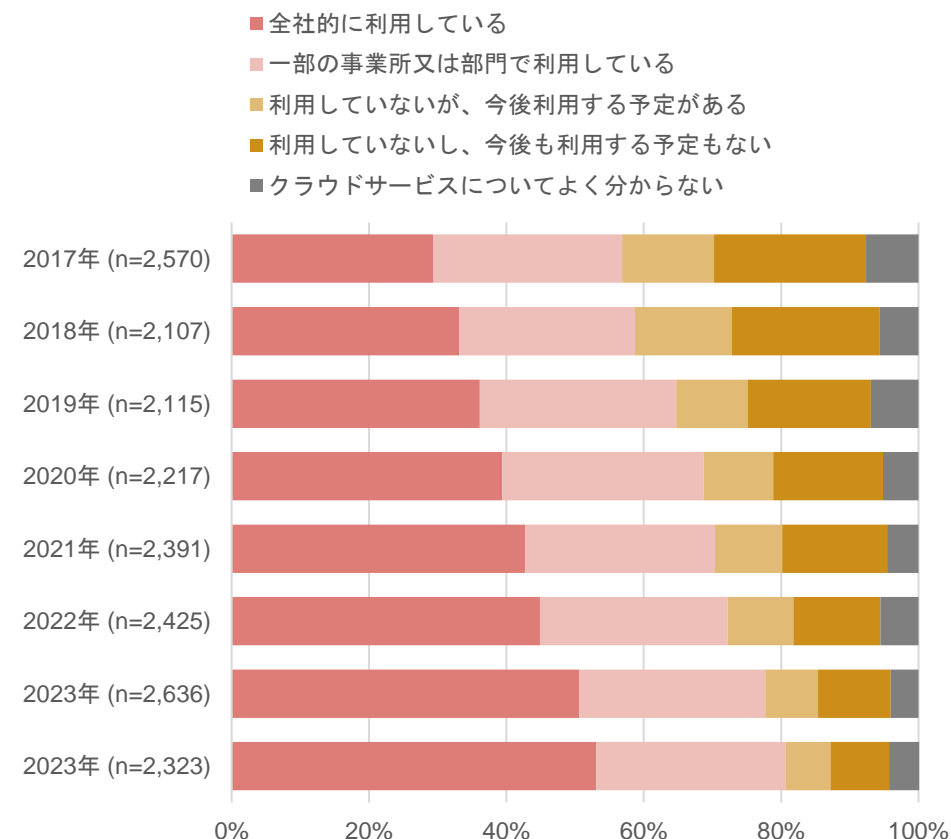


注：普及状況は、色が濃くなるに従って広範囲で普及している状態を示す
出所：各種資料より野村證券作成

企業のクラウド利用の現状

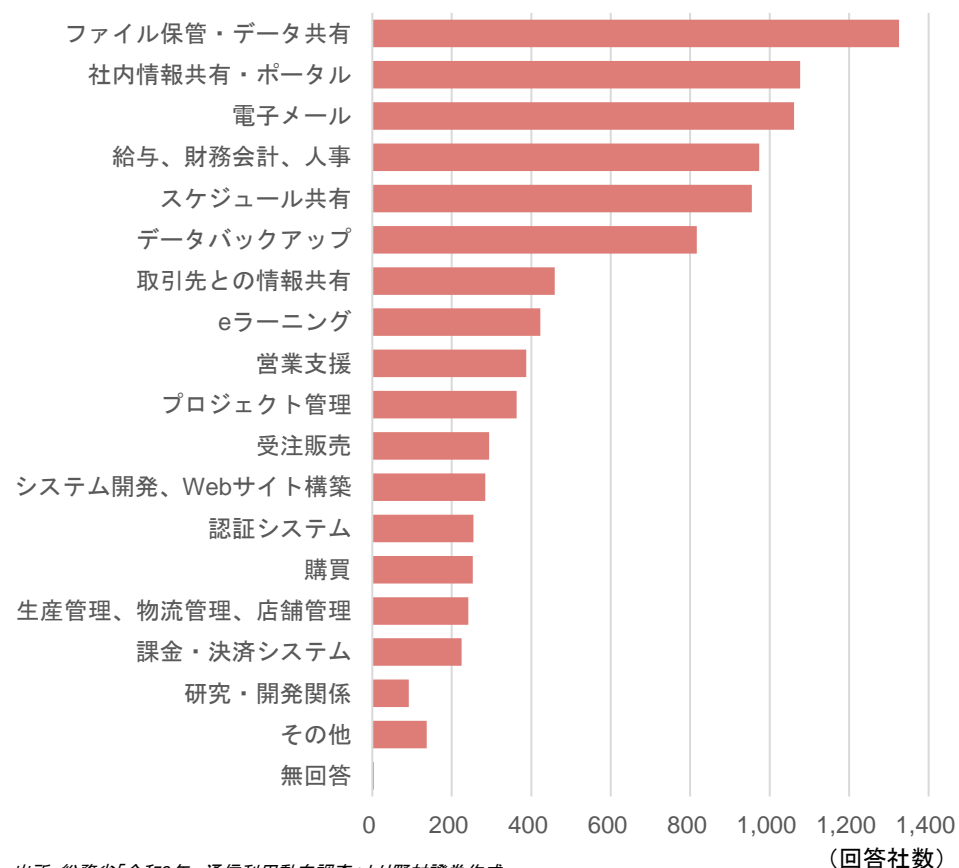
- SaaSやIaaS、PaaSといったクラウドサービスを利用する企業が増えている
- 多くの企業は、データの保存や共有、給与や会計等のバックオフィス業務に、クラウドサービスを利用している

クラウドサービスの利用状況



出所:総務省「令和6年 通信利用動向調査」より野村證券作成

企業が利用しているクラウドサービス(複数回答可)



出所:総務省「令和6年 通信利用動向調査」より野村證券作成

オンプレミス環境とクラウド環境の比較

- クラウド環境ではAPI連携を強化し易い。クラウドサービスプロバイダーがAPI連携サービスを豊富に揃えているほか、スケーラビリティが高く、連携する機能やサービスのコンピューティングリソースを増減させ易いためである
- オンプレミス環境はシステムをカスタマイズで設計できるほか、独自の情報セキュリティ対策を講じられる。そのため、システムや情報セキュリティの運用コストを最適化することは難しい

オンプレミス環境とクラウド環境の比較表

		オンプレミス環境	クラウド環境	
柔軟性	リソース増減の機動性 (スケーラビリティ)	低	高	
	API連携のし易さ (機能やサービスの拡充し易さ)	難 (システム改修や運用負担が高まる)	易 (標準化されたAPI連携サービスが豊富)	
	操作し易さ	難 (管理画面から手動で操作)	易 (プログラムで操作)	
経済性	システム構築時	カスタマイズ性	高 (自社で設計する)	低 (クラウド事業者が予め設計)
		初期コスト	高	低
	システム構築後	システム運用コスト	同程度	同程度
		リソース追加コスト	低	高
	コストの最適化し易さ		難	易
セキュリティ対策を講じる主体		自社	クラウド事業者 (マネージドサービス)	

ディスクレームー

本資料は、ご参考のために野村証券株式会社が独自に作成したものです。本資料に関する事項について貴社が意思決定を行う場合には、事前に貴社の弁護士、会計士、税理士等にご確認いただきますようお願い申し上げます。本資料は、新聞その他の情報メディアによる報道、民間調査機関等による各種刊行物、インターネットホームページ、有価証券報告書及びプレスリリース等の情報に基づいて作成しておりますが、野村証券株式会社はそれらの情報を、独自の検証を行うことなく、そのまま利用しており、その正確性及び完全性に関して責任を負うものではありません。また、本資料のいかなる部分も一切の権利は野村証券株式会社に属しており、電子的または機械的な方法を問わず、いかなる目的であれ、無断で複製または転送等を行わないようお願い致します。

当社で取り扱う商品等へのご投資には、各商品等に所定の手数料等(国内株式取引の場合は約定代金に対して最大1.43%(税込み)(20万円以下の場合、2,860円(税込み))の売買手数料、投資信託の場合は銘柄ごとに設定された購入時手数料(換金時手数料)および運用管理費用(信託報酬)等の諸経費、等)をご負担いただく場合があります。また、各商品等には価格の変動等による損失が生じるおそれがあります。商品ごとに手数料等およびリスクは異なりますので、当該商品等の契約締結前交付書面、上場有価証券等書面、目論見書、等をよくお読みください。

国内株式(国内REIT、国内ETF、国内ETN、国内インフラファンドを含む)の売買取引には、約定代金に対し最大1.43%(税込み)(20万円以下の場合、2,860円(税込み))の売買手数料をいただきます。国内株式を相対取引(募集等を含む)によりご購入いただく場合は、購入対価のみお支払いいただきます。ただし、相対取引による売買においても、お客様との合意に基づき、別途手数料をいただくことがあります。国内株式は株価の変動により損失が生じるおそれがあります。

国内REITは運用する不動産の価格や収益力の変動により損失が生じるおそれがあります。国内ETF・ETNは連動する指数等の変動により損失が生じるおそれがあります。国内インフラファンドは運用するインフラ資産等の価格や収益力の変動により損失が生じるおそれがあります。

外国株式の売買取引には、売買金額(現地約定金額に現地手数料と税金等を買いの場合には加え、売りの場合には差し引いた額)に対し最大1.045%(税込み)(売買代金が75万円以下の場合最大7,810円(税込み))の国内売買手数料をいただきます。外国の金融商品市場での現地手数料や税金等は国や地域により異なります。外国株式を相対取引(募集等を含む)によりご購入いただく場合は、購入対価のみお支払いいただきます。ただし、相対取引による売買においても、お客様との合意に基づき、別途手数料をいただくことがあります。外国株式は株価の変動および為替相場の変動等により損失が生じるおそれがあります。

野村証券株式会社

金融商品取引業者 関東財務局長(金商) 第142号

加入協会／日本証券業協会、一般社団法人 日本投資顧問業協会、一般社団法人 金融先物取引業協会、一般社団法人 第二種金融商品取引業協会