

SOLUTION BRIEF

マイクロサービスのための API アーキテクト・ガイド

# マイクロサービスのための アーキテクト・ガイド

CA Technologies のツールによるマイクロサービスと  
API 開発の加速

CA Technologies はスケーラブルで安全なマイクロサービス・ソリューションのための実績あるプラットフォームを企業に提供します。

# 概要

## 課題

多くの企業はアプリケーション・エコノミーに直面して、変化するニーズに迅速に適応しなければならないアプリケーションを再構築する必要に迫られています。大規模なアプリケーションをロールアウト（およびサポート）するために、従来の方法は十分ではありません。現在の企業のアプリケーションのアーキテクトと担当 VP は以下のような問題を抱えています。

- モダンなアプリケーションを数か月間や数年間ではなく数日間または数週間でデプロイ / リリースするにはどうすればよいか？ また、アプリケーションの更新に伴うダウンタイムを最小限に抑えるにはどうすればよいか？
- こうしたモダンなアプリケーションを構築するために、異なる言語のプラットフォーム上で複数の開発チームを活用するにはどうすればよいか？
- ニーズの変化に応じてアプリケーションを拡張し、同時にその拡張に適応するためのインフラストラクチャ・コストを最小限に抑えるにはどうすればよいか？

## ビジネス・チャンス

これらの課題は、モダンなアプリケーションを構築するためにアジリティと規模を重視する傾向が増したことに端を発しており、従来のアプリケーション開発手法ではこの環境に対応できません。

CA Technologies はフル・ライフサイクル API 管理にマイクロサービスを含めるよう拡大しました。この統合によって最善の組み合わせの機能が連動でき、モダンなアーキテクチャにプラットフォームを提供し、アジリティと規模のための環境を保護できます。CA を使用することで企業はベスト・プラクティスと業界をリードするテクノロジーを活用でき、アーキテクチャのモダナイゼーション・プロセスを加速し、もっと実用的なものにできます。

## メリット

CA Technologies はマイクロサービス・アーキテクチャ向けの設計、開発、デプロイ、監視、および管理のプラットフォームを提供します。

- **CA Live API Creator** はマイクロサービスの設計とデプロイのための使いやすいソリューションを提供し、モダンなアプリケーションのタイムツーマーケットを短縮します。
- **CA Microgateway** と **CA API Gateway** はマイクロサービス・アーキテクチャが必要とするセキュリティ・インフラストラクチャを提供し、エンタープライズ IP を確実に保護します。
- **OAuth Toolkit** は CA API Gateway のコンポーネントで、アイデンティティ / アクセス・マネジメント・ソリューションの認証と統合を可能にし、アプリケーション・ユーザの CX を改善し、エンタープライズ・セキュリティをエンドツーエンドに拡張します。
- **CA Application Performance Management** は、コンテナ化されたシステムを監視し、アプリケーションの可用性と優れた CX をプロアクティブに確保します。

## セクション 1 :

# マイクロサービスとデジタル・トランスフォーメーション

本書では、企業がアプリケーション・エコノミーのニーズに対応するために必要なアジリティを、マイクロサービスがどのようにして提供できるかについて説明します。デジタル・トランスフォーメーションは簡単なことではありません。実際、モダンなアプリケーション・アーキテクチャの構築は設定と実装が非常に骨の折れる仕事です。本書では以下のような疑問に答えます。

- マイクロサービスとは何か？
- マイクロサービスはなぜそんなに重要なのか？
- マイクロサービスは本当に必要か？
- マイクロサービスを推進するにはどうすべきか？
- マイクロサービスと API の接続はどのように行うべきか？
- マイクロサービスを保護するにはどうすべきか？

## マイクロサービスとは何か？

現在のデジタル・エコノミーでは、単一のアプリケーションはもはや効率的ではありません。どの企業の開発チームも、大規模なアプリケーションの開発時間、デプロイ、スケーラビリティの問題に対処しようと努力しています。DevOps フレームワークとアジャイル開発が広く普及するにつれて、開発チームは複雑なアプリケーション・サイロを単純なコード・ブロックに分割する必要性を痛切に感じるようになり、マイクロサービスが誕生することになりました。

アーキテクトは、複雑なアプリケーションを、簡単できめ細かく再利用や相互運用が可能なプロセスに変える設計パターンを考案しました。これらのプロセスはそれぞれ個別に変更およびデプロイが可能です。これらのきめ細かいプロセスはマイクロサービスと呼ばれます。

## マイクロサービス・コンポーネント

[API アカデミー](#)はマイクロサービス・アーキテクチャの特定の共通した性質を把握しており、業界はこうした性質を標準として採用してきました。このことを考慮すると、マイクロサービスは以下のような重要な性質を有しています。

- 小規模
- メッセージングに対応
- コンテキストによる制限
- 自律的に開発
- 個別にデプロイ可能
- 分散
- 言語中立
- 自動プロセスによる構築とリリース

マイクロサービスとマイクロサービス・アーキテクチャという言葉が代わる代わる使用されますが、これらは同じではありません。マイクロサービス・アーキテクチャは高度に自動化された発展可能なソフトウェア・システムで、機能に応じた複数のマイクロサービスで構築されています。

一般に、これらのコンポーネントは [Docker コンテナ](#) にデプロイされ、一元化されたランタイム環境を提供します (ただし多数の Docker コンテナを持つことが可能)。Docker コンテナを使用する別のメリットは、管理の労力をいくらか減らせることです。

### マイクロサービスはなぜそんなに重要なのか？

デジタル・エコノミーでの成功を目指すデジタル・エンタープライズはすべて、スピードと規模という2つを追求しています。会社にとって市場投入の迅速化に対するニーズが重要であるなら、増大する顧客のニーズをサポートするために適切に拡張できることもまた同様に重要です。しかし、ここでのモットーは、さらに大きな規模でのスピードと安全です。安全を犠牲にすることなくスピードと規模を達成できなければ成功はありません。

アジャイルおよび DevOps モデルはソフトウェア資産の分散所有をサポートしており、より迅速な変更のターンアラウンドと迅速なデプロイを促進します。ただし、複雑な単一のアプリケーションを自律したユニットにインテリジェントに分割するためには、設計戦略、つまりマイクロサービスが必要です。巨大なアプリケーションをマイクロサービスに分割すると、開発チームが更新や自律的デプロイをより機敏に行えるようになります。その結果、大規模で複雑なビルドを作成するための依存関係が取り除かれ、ボリュームのニーズに合わせてスケーラビリティを向上させるために複雑すぎるアーキテクチャは不要になります。

### 当社はマイクロサービスからメリットを得られるか？

すべての企業にはそれぞれ異なるニーズがあります。マイクロサービスがソリューション提供の優れた方法である場合もあれば、単一のアプリケーションがより適切な場合もあるでしょう。

単一のアプリケーションには以下のような特徴があります。

- 一般的に、最初は構築しやすい
- (一般的に) 最初はテストを行いやすい
- 統合開発環境 (IDE) を使用している場合は、より適合しやすい

ただし単一のアプリケーションには以下のような特徴もあります。

- コードベースが拡大すると、維持管理が難しい
- コードベースの拡大により、時間の経過とともに反復にかかる時間が長くなる
- 拡張が困難で、スケーラビリティに問題があるのが一つのアプリケーション・コンポーネントのみであっても、インフラストラクチャは一般的にアプリケーション全体のために規模を拡大する必要がある
- 根本にある特定のコードベースのために、革新がより困難
- 容易に習熟できるため (および根本の特定のコードベースのため)、新しいプログラムを投入するのが困難

こうした理由で、単一のアプリケーションはアジャイルの実現や適切に拡張を行うことが非常に困難な場合があります。しかし既存の単一の環境があり、アジリティを必要としない場合、マイクロサービスベースのソリューションは重要ではないかもしれません。

### マイクロサービスを実装する場合、何を得られ、どのようなコストがかかるか？

マイクロサービス・アーキテクチャを実装することは、アプリケーション・エコノミーにおいていくつかのメリットをもたらします。

マイクロサービスベースのアプリケーションには以下のような特徴があります。

- 大規模なアプリケーションに適したアーキテクチャ: コンポーネントを希望どおりに (アプリケーション全体に影響を及ぼさずに) 構築 / 交換できます。また、何か問題が起きた場合も、影響を受けるのはそのコンポーネントだけです。
- よりアジャイルなソリューション: 必要に応じて容易にアプリケーションのセグメントの方向性を転換させられます。

- 習熟が容易：各コンポーネントは小規模で分離しているため、各コンポーネントが何をどのように実行するかを判断したり、アプリケーションとのインタラクションについて判断したりするプロセスがより簡単です。
- 拡張が容易：アプリケーション全体が影響を受けるのではなく、スケーラビリティの向上を必要とするコンポーネントのみが影響を受けます（企業にとって大幅なコスト削減にもなります）。

ただし、マイクロサービスベースのアプリケーションには以下のような特徴もあります。

- 単一のアプリケーションよりも可動部分が多いため、監視が重要
- 拡大するコードベースに対しては、よりモダンなバックエンド・インフラストラクチャ要件
- アプリケーション全体のテストがより困難（ただし個別のコンポーネントのテストは大幅に容易）
- エンドポイントに代わってセキュリティ要件に対応することが必要

デジタル・トランスフォーメーション・ジャーニーが進行中でアジャイルを推進する必要がある企業にとっては通常、マイクロサービスのメリットはデメリットをはるかに上回ります。

### 社内で方法マイクロサービスを推進するにはどうすべきか？

**アジャイルな企業の成熟度を評価する：**組織のアジャイル化が実現しており、DevOps を検討中か採用済みの場合、その組織はマイクロサービスの投入準備が整っています。

**少人数の開発者グループを作成する：**少人数の開発者チームが小規模なサービス / API のセットを所有して効果的に作業できるようにします。その結果、柔軟性の高い結合と自律的なデプロイが促進されます。

**ドメイン中心の設計を採用する：**大規模なアプリケーションを、ビジネス機能に基づいてより単純なサービスに分割します。サービスがきめ細くなるほど、この設計の効果が期待できます。

### API とマイクロサービスの接続はどのように行うべきか？

マイクロサービス・コンポーネントは、システム内の他のコンポーネントとコミュニケーションができて初めて貴重な存在になります。そのため、各マイクロサービス・コンポーネントはインタフェースまたは API を備えています。コードに高レベルの分離、独立性、およびモジュール性が必要であると同様に、コンポーネントのインタフェースである API 間も高い柔軟性で結合している必要があります。さもないと、スピードと安全のバランスをとるための主な目標の 1 つである、2 つのマイクロサービスを個々にデプロイすることができなくなります。

マイクロサービスの前にある API 層はきめ細かいマイクロサービスをアプリケーションから切り離すため、クライアント側のアプリケーション(モバイルなど)のサポートが容易になります。この層はマイクロサービス・オーケストレーションの実行とセキュリティの適用に最適です。

### マイクロサービスを保護するにはどうすべきか？

マイクロサービスの事実上すべての実装で一般的に見られるのが、マイクロサービスが提供する API エンドポイントをチームが API ゲートウェイで保護するというパターンです。現在の API ゲートウェイは、マイクロサービスに求められる重要な機能、すなわちトランスフォーメーションとオーケストレーションを追加で提供します。最後に、最も成熟した実装では、API ゲートウェイがサービス検出ツールと連携してマイクロサービス・クライアントからの要求をルーティングします。マイクロサービス・アーキテクチャは、非常に自由度の高いアーキテクチャです。複雑なエンタープライズ・アプリケーション用にアーキテクチャが実装された成熟したマイクロサービス組織では、一般に何百ものマイクロサービスがデプロイされます。このケースでは、セキュリティが非常に重要な要素になります。実際、マイクロサービスのすべての実装で、優れた API ゲートウェイを使用して保護されたさまざまなマイクロサービスが提供する API エンドポイントが見られます。マイクロサービスが提供する API は、互いを呼び出すこともあれば、フロントエンドの（公開された）API によって呼び出されたり、モバイル・アプリケーション、Web アプリケーション、パートナー・システムなどの API クライアントによって直接呼び出されたりすることもあります。優れた API ゲートウェイと OAuth プロバイダを組み合わせ使用して、マイクロサービス対応システムの公開された API エンドポイントの呼び出しを保護するアプローチが広く推奨されています。API ゲートウェイは、

マイクロサービス・アーキテクチャの鍵となるコンポーネントであり、実装されたサービスとそれを利用するクライアントとの橋渡しを行います。

API ゲートウェイは以下を提供します。

- 認証、権限認証、脅威からの保護を目的とした一元的なセキュリティの適用
- さまざまなプロトコルにまたがる保護されたリソースへのルーティングと仲介
- ビジネス・レベルのレート制限およびクォータを適用するためのサービスレベル管理
- サービスの呼び出しを減らすためのサービス・オーケストレーション
- 単一のバックエンドからアプリケーション固有のインタフェースを公開するための、サービスのファサード

ゲートウェイはマイクロサービスに代ってセキュリティを適用します。ただし通常は、セキュリティ・セッションを管理し、API ゲートウェイと連携して機能する OAuth プロバイダもあります。

## セクション 2 :

# マイクロサービス用の CA Technologies のツール

前述のように、マイクロサービス・アーキテクチャは可動部分が**多く**、すべてがシームレスに連動する必要があります。CA Technologies はこれを包括的な方法で表示し、総合的ソリューションを提供するプラットフォームとしてフル・ライフサイクル API 管理機能を提供します。

**CA Live API Creator** は CA の自動化されたロー・コード代替開発ソリューションです。レガシーおよび現在のデータ・ソースおよびアプリケーションから既存のデータおよび機能へのアクセスを提供するために、ドメイン・ドリブンのマイクロサービスと、アプリケーションのバックエンドとしての REST/JSON API を作成して公開します。これにより、開発者はポイント・アンド・クリック・アプローチを使用して、多様なデータ・ソースからのデータを結合する新規の REST エンドポイントを作成できます。API オーナは、宣言型ビジネス・ルール、JavaScript イベント処理、役割ベースのセキュリティ、および対話型テストによって、API を拡張できます。また、CA Live API Creator は、API 管理を採用した企業がその API ライフサイクルの範囲を、既存のゲートウェイおよびポータル製品における管理および適用の範囲を超えてよりデータ・レイヤに近い API の作成へ向けて拡張することも可能にします。CA Live API Creator を使用すれば、内部のアプリケーション、モバイル開発プロジェクト、サービスとしてのデータ (DaaS) の公開、IoT (Internet of Things) の有効化、およびパートナーとの統合のために、アプリケーション・バックエンドを迅速に作成できます。

マイクロサービスの観点から見て、このソリューションはいくつかの重要な目標の達成に役立ちます。

- **モジュール性** : CA Live API Creator を使用して大規模なアプリケーションをリソースと呼ばれる自己完結型のユニットに分解することで、アプリケーションのデリバリーに必要なすべてのもの、すなわちデータの統合、ビジネス・ロジック、および堅牢な API 相互作用レイヤを提供します。リソースは、その土台にあるスキーマに依存しないメッセージベースの RESTful API です。
- **デリバリーの迅速性** : リソース定義では、ポイント・アンド・クリックによって複数のデータベースからの複数のオブジェクトを統合します。
- **ゼロ・デプロイ** : このソリューションは、デプロイに関連する遅延を排除します。定義されたリソースは、保存されるとすぐに実行可能となり、コンパイルもデプロイも不要です。
- **自動デプロイ** : 代わりに、マイクロサービスをエクスポートし (たとえば開発から)、スクリプトを使用してインポートする (たとえば本番環境へ) こともできます。

- ・**まとまり** : 依存関係が自動化されるため、1つのマイクロサービスをデプロイしても他のマイクロサービスに影響しません。
- ・**懸念事項の分離** : CA Live API Creator は、マイクロサービスの作成をビジネス・ロジックから切り離します。ビジネス・ロジックは、土台のドメイン上で定義されます。

**CA Microgateway** はコンテナ化された軽量のゲートウェイで、高度に分散化された環境で拡張できるよう設計されています。サービスの検出、ルーティング、レート制限、ラスト・マイル・セキュリティ、ローカルでの集約とオーケストレーションなどの機能を提供し、一般的なマイクロサービスのパターンに対応します。また付属のポリシー・テンプレートを使用して、設計時に開発者が容易にデプロイおよび設定できます。業界標準の DevOps ツールと統合し、スクリプト作成された本番のデプロイに対応します。また、新しいテンプレートを作成して新規の事前定義されたコンテナにベキングすることで、カスタム / 新規ユースケースをサポートするように拡張できます。

CA Microgateway では、マイクロサービス環境で新たに登場した新規パターンに開発者が対応できます。また、規模の大小を問わずマイクロサービス・アーキテクチャに必要なトラフィック管理とメディエーションの機能を提供します。さらに、業界規制への対応に必要なセキュリティと耐障害性も備えています。

**CA API Gateway** (Essentials、Enterprise、および CA Mobile API Gateway) は、ポリシー管理機能とランタイム・ポリシー適用機能を組み合わせ、場所にかかわらず企業とエンドユーザの間に中央のポリシー適用ポイントを提供することにより、エンタープライズ規模のマイクロサービスに対応した業界屈指のゲートウェイ機能を提供します。CA API Gateway を使用すれば、企業は自社のデータやアプリケーションを、社内およびサードパーティの開発者の両方に対して選択的に開放することができ、既存の IAM (アイデンティティ / アクセス管理) ソリューションと統合してプラグ & プレイ・ソリューションを構築できます。CA API Gateway は、さまざまなフォーム・ファクタでデプロイされます。これには、容易に規模を拡大でき、フェイルオーバー環境にデプロイして高可用性を得ることができるためマイクロサービスにとって理想的な Docker も含まれます。CA API Gateway にはプロトコル・ブリッジング機能もあり、さまざまなプロトコル間の完全な変換 (レガシーから REST や JSON へ) が可能で、レガシーからモバイル、クラウドおよびソーシャルへのブリッジングを行えます。

マイクロサービス・アーキテクチャのデプロイの観点から見て、CA Microgateway と CA API Gateway は次のようにいくつかの重要な目標の達成に役立ちます。

- ・**セキュリティ** : 前述のように、CA API Gateway は中央のポリシー適用ポイントとして機能できます。API / マイクロサービスへのあらゆるアクセスを API ゲートウェイによって常に保護する方が良策であり、ほとんどのケースでは、サービス・コール間に API ゲートウェイを導入するためにかかるオーバーヘッドは無視できるほど小さく、その利点には十分な価値があります。
- ・**トランスフォーミングとオーケストレーション** : CA API Gateway を使用すると、構成を通じて宣言によって API インタフェースを作成できます。API インタフェースは、バックエンドのマイクロサービスを調整してその細かさを開発者が使いやすいインタフェースの背後に「隠して」おしゃべりを排除できます。
- ・**ルーティング** : CA API Gateway は、クライアント・アプリケーションからマイクロサービスへのルーティングの複雑さを見えなくします。このソリューションは、サービス検出システムの HTTP インタフェースまたは DNS インタフェースのどちらとでもインタフェースでき、マイクロサービスに関連付けられた外部 URI が要求されたときに API クライアントを正しいサービスへとルーティングします。

**CA Microgateway は、同じ PaaS コンテナ管理環境内で、管理対象のマイクロサービスと連携してデプロイおよび拡張するために最適です。**多くは CA API Gateway と組み合わせてアプリケーション階層またはネットワークのエッジにデプロイしたり、OAuth サーバや一元的なログ記録 / 監査システムなど既存のインフラストラクチャと統合したりして使用します。CA API Gateway はマイクロサービス環境で CA Microgateway と組み合わせるのが理想的ですが、従来の単一アーキテクチャおよび API 中心の (マイクロサービスではない) アーキテクチャにも適用可能です。



CA の **OAuth Toolkit** は、業界最高の CA API Gateway 上で実行します。OAuth Toolkit は OAuth プロバイダと、Web、モバイル、その他のアプリケーションからマイクロサービスへのアクセスを制御するトークン管理システムを提供します。OAuth Toolkit は既存のアイデンティティ・インフラストラクチャを拡張することで、これらの OAuth プロバイダ機能を提供できるようにし、高い拡張性も備えています。このソリューションには、以下が含まれます。

- 2-legged、3-legged の両 OAuth フローでアクセス・トークンを発行する OAuth 認可サーバ
- API のアクセス制御とポリシー適用を行う OAuth リソース・サーバ
- OAuth クライアント実装、ユーザ実装のためのカスタマイズ可能なテンプレート
- 一般的なあらゆる IAM とシングル・サインオン (SSO) ソリューションと統合
- OAuth と他のアクセス制御の業界標準との連携機能
- JWT などのトークン・タイプの中から選択可能
- カスタマイズされたユーザ・エクスペリエンス向けにカスタム・ハンドシェイクの実装が可能

CA の OAuth Toolkit を使用することで、マイクロサービス用の分散認証メカニズムを構築でき、ソリューションの安全性を確保できます。

**CA Application Performance Management (CA APM)** は動的なマイクロサービスと、コンテナの一時的な性質を管理するための、独自のアーキテクチャを提供します。従来の静的なトポロジ・マッピングと、単一のシステムに最適なインスツルメンテーションは、マイクロサービスにとっては関連性が低いということを認識し、CA APM はコンテナ化されたシステムを管理するために徹底して未来に対応できるアプローチを使用しています。このアプローチの基盤となっているのは、迅速で簡略化された設定と、モダンなシステムの複雑性、特にマイクロサービスの相互依存性と通信フローの可視化です。

マイクロサービス・アーキテクチャ向けの CA APM は、多面的な監視ソリューションです。基本的なサービスとして、エージェントレス監視によってコンテナと依存関係の検出を自動化し、CPU 飽和度、エラー率、遅延など主要なヘルス・インジケータを即座に表示します。このソリューションはそれ自体がパワフルなサービスですが、コンテナ特性の自動キャプチャと、マイクロサービスの性能を複数の観点から表示できるデータ・モデルによって、さらに強化されています。このアプローチは、エンジニアが迅速かつ容易に複雑なトポロジをサービス・ビューに抽出でき、そこで性能が自動的に収集されるため、マイクロサービス・アーキテクチャに非常に適しています。

多くの場合、コンテナ監視はアプリケーション中心の性能インジケータによって強化する必要があります。CA APM は、コンテナ内のアプリケーション・インスツルメンテーションを有効にすることで、これをサポートします。これによって、マイクロサービス・アーキテクチャのサポートする際に、高度なアプリケーション性能サービスへのアクセスを可能にします。たとえば、統計的技術は性能のベースライン化を管理してアラートのノイズ減らすことができ、また、トランザクション追跡機能と優先順位判定支援機能は詳細な証拠を収集でき、修復ワークフローを構築します。

エージェントレスのコンテナ中心監視機能と、さらに詳細なアプリケーション・インスツルメンテーションはそれ自体が価値あるサービスですが、CA APM はそれらが示す情報を統合し、高レベルな洞察を提供します。CA APM はアプリケーション性能をコンテナのヘルスと自動的に相関させることで、問題の正確な根本原因を示す機能を DevOps チームに提供するだけでなく、どのコンテナ・アプリケーション設定が最高の性能を実現できるかについて詳細を示します。

## セクション 3:

## 次のステップ

DevOps とアジャイル開発を導入した今日の企業は、迅速な変更への対応とすばやいデプロイの実現に努力しています。このような企業にとって、マイクロサービス・アーキテクチャは恩恵をもたらしますが、万能の特効薬ではありません。組織はより小規模な開発チームにより高い自律性と機敏性を持たせることができ、その結果として、企業の IT 部門がニーズの変化に適切に対応できるようになります。

IT 部門はその API 戦略を、開発者が生み出すマイクロサービスに合わせる必要があります。これらのマイクロサービスの保護は最も重要な課題ですが、それには API ゲートウェイの活用が有効です。さらに、スピードと規模を追求する場合は、安全も等しく重要であることを忘れないでください。また、強力な管理コンポーネントも必要です。

マイクロサービスの詳細については、以下の弊社の eBook をダウンロードしてください。

[\[Microservice Architecture: Aligning Principles, Practices, and Culture\]](#)

CA API Management の詳細については、[ca.com/jp/microservices](https://ca.com/jp/microservices) にアクセスしてください。



[ca.com/jp](https://ca.com/jp)でCA Technologiesにアクセスしてください。



CA Technologies (NASDAQ : CA) は、企業の変革を推進するソフトウェアを作成し、アプリケーション・エコノミーにおいて企業がビジネス・チャンスを獲得できるよう支援します。ソフトウェアはあらゆる業界であらゆるビジネスの中核を担っています。プランニングから開発、管理、セキュリティまで、CA は世界中の企業と協力し、モバイル、プライベート・クラウドやパブリック・クラウド、分散環境、メインフレーム環境にわたって、人々の生活やビジネス、コミュニケーションの方法に変化をもたらしています。詳細については [ca.com/jp](https://ca.com/jp) をご覧ください。