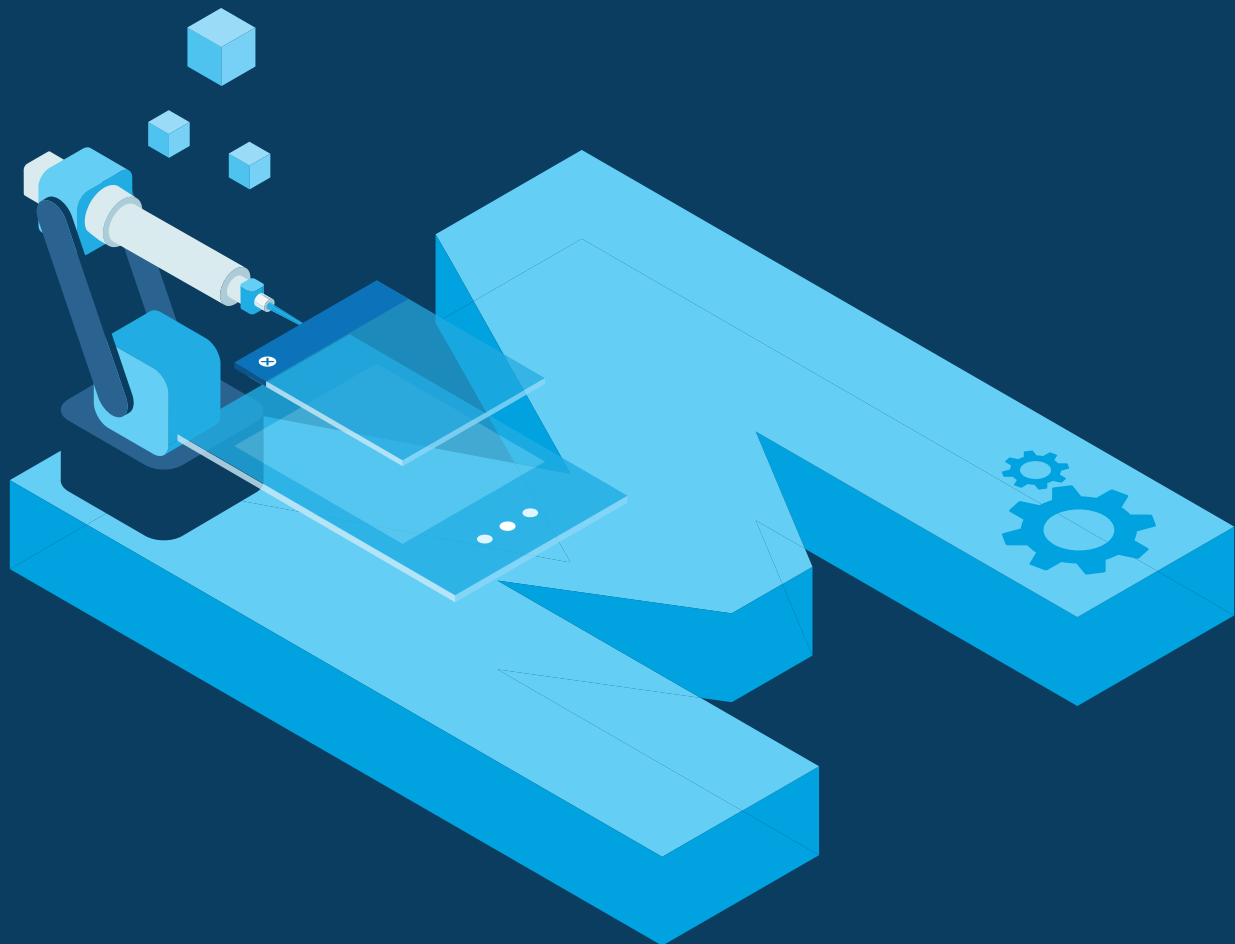




モダンなアプリケーション開発を実現する

マイクロサービス入門ガイド



目次

1

マイクロサービスと App Cloud

2

マイクロサービスの基本アプローチ

2.1 データオブジェクトマイクロサービス

2.2 プロセス駆動型マイクロサービス

2.3 カスタムマイクロサービス

2.4 すぐれたマイクロサービス設計の構成要素

3

マイクロサービスの種類

3.1 データオブジェクトマイクロサービス

3.1.1 ネイティブ Salesforce オブジェクト

3.1.2 外部データオブジェクト

3.1.3 データオブジェクトの API 公開の自動化

3.2 プロセス駆動型マイクロサービス

3.2.1 データオブジェクトの API 公開の自動化

3.2.2 Lightning プロセスビルダーで作成されるビジュアルプロセス

3.3 カスタムマイクロサービス

3.3.1 任意の言語を使用し、Heroku でデプロイ

3.3.2 Apex で記述された、アノテーション付きのオブジェクト指向クラスとメソッドを使用

3.3.3 アクションベースのカスタムマイクロサービス

4

まとめ

1. マイクロサービスと App Cloud

開発パターンの最旬トレンドとして、瞬く間に認知されるようになったマイクロサービス。これは単に、より良い開発を行うためのサービス指向アーキテクチャ（SOA）の進化形と考えてよいのでしょうか？

マイクロサービスは、コードの独立・分離を促進するための、SOA にもとづく開発手法と最先端のソフトウェアデザインとの中間に位置するものと捉えられているようです。しかし、ビジネスの成果につながるソリューションをすばやく提供することの大切さを理解している現代の開発者にとって、その定義自体はさほど重要ではありません。ビジネスのニーズを満たすには、アーキテクチャやインフラストラクチャをどうするかにとらわれている時間はないのです。

App Cloud

ビジネスの成果につながるソリューションの実現を可能にするのが、Salesforce App Cloud です。強力なロールベースのセキュリティ、マネージドインフラストラクチャ、自動で利用可能となる API、サービスのバージョン管理などを提供するマルチテナント型の PaaS（サービスとしてのプラットフォーム）をご利用いただけます。

また、アプリケーション開発とデータ管理を高速に行えるビジュアルツールも提供しています。

マイクロサービスのアプローチを採用し、App Cloud を基盤とすることで、ビジネスのニーズに応えるサービスを驚くほど短期間でリリースできます。



2. マイクロサービスの基本アプローチ

PaaS の利用によるインフラストラクチャ管理からの解放は、マイクロサービスアーキテクチャによるアプリケーション開発の第一歩です。マイクロサービスには、次の 3 つの基本アプローチがあります。



2.1 データオブジェクト マイクロサービス

ビジネスユーザーや開発者は、App Cloud の直感的なビジュアルツールを使用して、データベースやさまざまなタスクを実行するアクションをクリックするだけで作成できます。作成した機能は、マイクロサービスとして自動で公開できます。こうしたサービスは REST ベースのエンドポイントとして、サードパーティの開発者にも公開されます。



2.2 プロセス駆動型 マイクロサービス

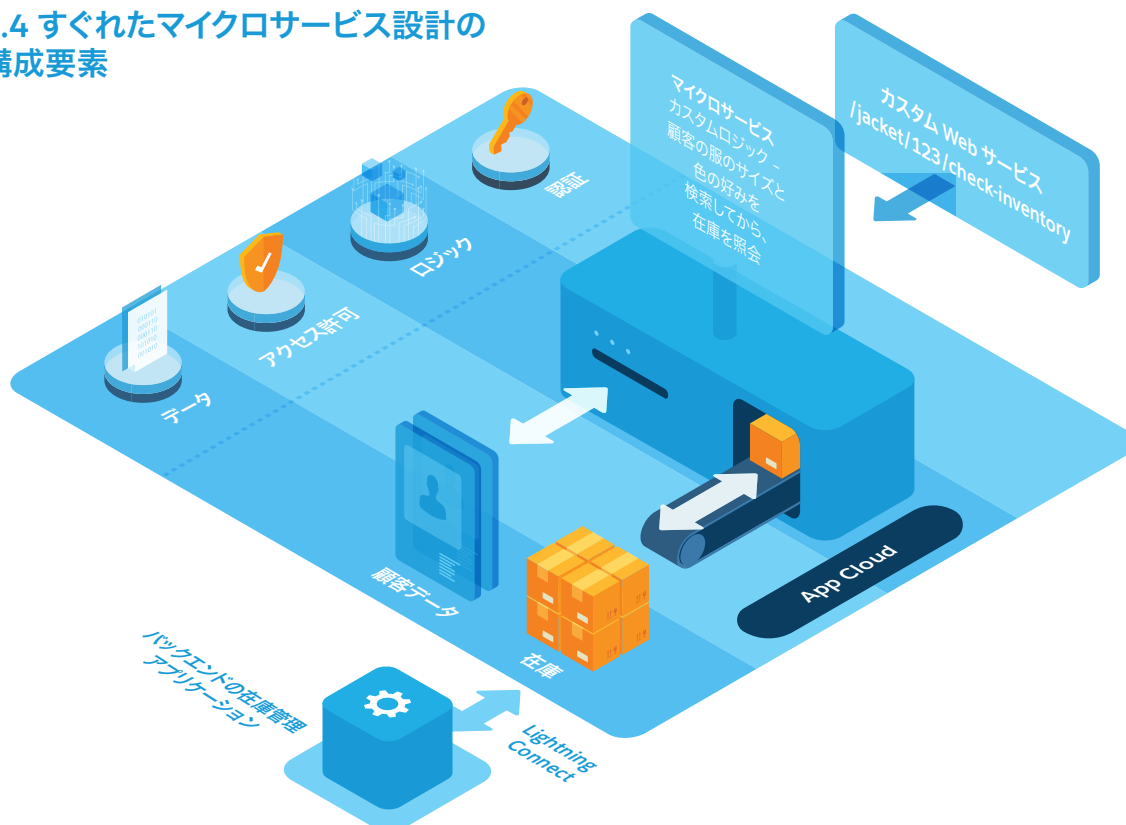
プロセス駆動型マイクロサービスでは、データオブジェクトマイクロサービスとカスタムマイクロサービスの両方をエントリーポイントとして、App Cloud のコア機能である Lightning プロセスビルダーで定義した各種のプロセスを実行できます。オブジェクトの作成と更新をカプセル化し、独立したマイクロサービスにすることで、より効果的な設計が可能になります。



2.3 カスタム マイクロサービス

開発者はカスタムロジックのほか、データの変換、外部システムの呼び出し、トランザクションの管理のためのコードを、複雑なミドルウェアを介さずに任意の言語で作成し、マイクロサービスとして公開できます。次のページに、App Cloud を基盤としたマイクロサービスアーキテクチャの例を示します。

2.4 すぐれたマイクロサービス設計の構成要素



どのアプローチを選ぶかにかかわらず、マイクロサービスの開発では、SOA の基本理念を前提にする必要があります。つまり、あらゆるデータ利用者を想定して設計された汎用的なサービスにありがちな複雑さを解消し、それぞれに独立したサービスを通じて情報にアクセスできるようにすることが重要です。これを達成するため、マイクロサービスでは次の設計原則に従うことをお勧めします。

- 1 マイクロサービスは単一の機能だけを実行する
- 2 マイクロサービスは個別のデータエンティティまたは個別の機能を提供し、同一サービス内に両方を共存させない (1 つ目の原則にもとづく)
- 3 マイクロサービスは特別なミドルウェアの使用を必要としない
- 4 カスタムマイクロサービスは目的のタスクに応じたもっとも効率的な言語で記述し、特定の環境の使用を義務付けない
- 5 マイクロサービスはデータアクセスの方法を統一し、疎結合および契約による設計パターンを推進する

3. マイクロサービスの種類

ここでは、データオブジェクトマイクロサービス、プロセス駆動型マイクロサービス、カスタムマイクロサービスについて詳しく解説し、それぞれの基本的な違いを説明します。

3.1 データオブジェクトマイクロサービス

データオブジェクトマイクロサービスの主な機能は、単一の種類のデータ（注文、顧客、商品など）を提供することです。一般的なエンタープライズシステムの多くは、クラウドベースのアプリケーション、オンプレミスシステム、ERPといった複数のソースのデータを集約します。その結果、マイクロサービス環境内でのデータの連携に時間を取られ、保守性が低下する原因になります。App Cloud では、データエンティティをオブジェクトとして表現し、開発者がプログラム型や宣言型のツールから透過的にアクセスできるようにすることでこの問題を解決しています。これらのオブジェクトを使用し、一貫性のあるインターフェースを提供することにより、データのソースを問わずにデータをやり取りできます。次に、サポートされるオブジェクトの種類について説明します。

3.1.1 ネイティブ Salesforce オブジェクト



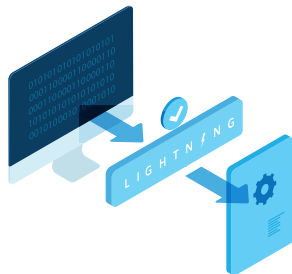
ネイティブ Salesforce オブジェクトには、Salesforce で提供されるデータエンティティ（取引先、取引先責任者、リード、キャンペーンなど、Salesforce 製品で使用されるエンティティ）と、スキーマビルダーなどの宣言型ツールを使用してユーザーが作成するデータエンティティがあります。



データベースの設計や管理に関する知識がなくても簡単に作成できます。

Salesforce で提供されるオブジェクトは「標準オブジェクト」、ユーザーが作成したオブジェクトは「カスタムオブジェクト」と呼ばれます。

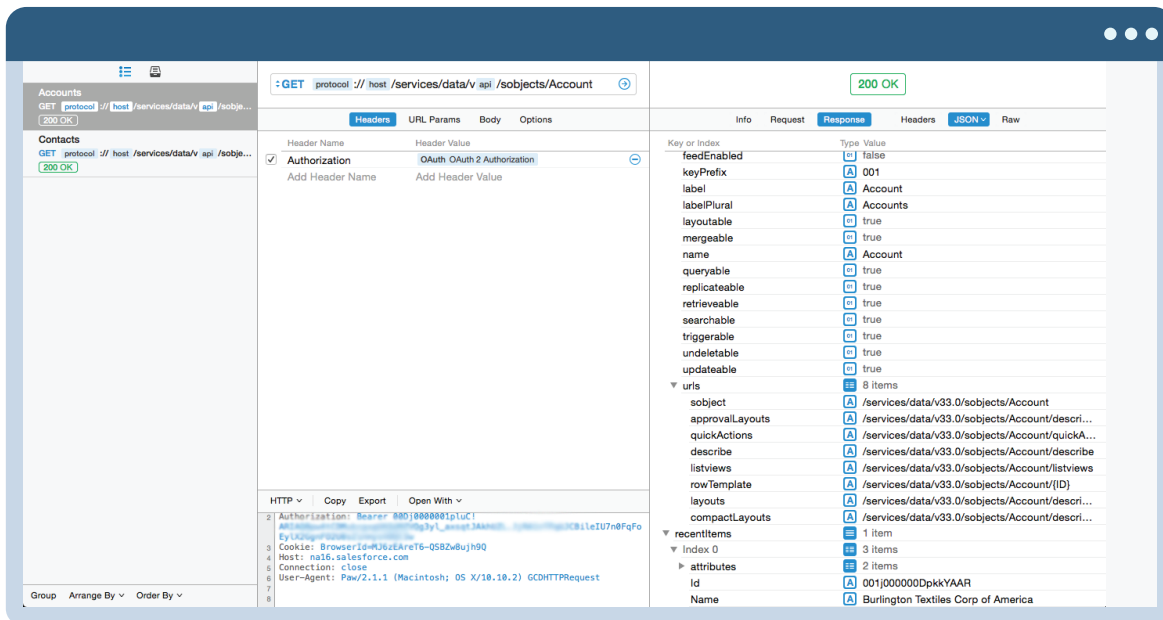
3.1.2 外部データオブジェクト



従来の連携方法では、オンプレミスアプリケーションのデータにアクセスするには ETL（抽出、変換、ロード）ツールが欠かせませんでした。一方、App Cloud の機能である Salesforce Connect では、OData ベースのアプローチを採用し、実行時に同期データの検証を行います。この同期データはデータオブジェクトとして扱われ、外部オブジェクトと呼ばれます。外部データオブジェクト構造が同期されると、ネイティブ Salesforce オブジェクトと同様に機能するようになります。

3.1.3 データオブジェクトの API 公開の自動化

App Cloud のデータオブジェクトは、単なる構造化データではありません。データオブジェクトは、関連付け、式を使った項目、住所の位置情報をサポートするほか、API 経由のアクセスに完全対応し、CRUD 操作を実行できます。これらの API は、Salesforce 独自の言語とフレームワーク（Apex、Visualforce、Lightning コンポーネントなど）のほか、一般的な JavaScript フレームワーク（Angular、Ionic、Backbone など）のいずれかを使用して App Cloud 上でアプリケーションを構築する開発者に自動的に公開されます。また、App Cloud では、こうしたデータオブジェクトが REST ベースのマイクロサービスとして公開されるため、あらゆる場所からの接続が可能です。



3.2 プロセス駆動型マイクロサービス

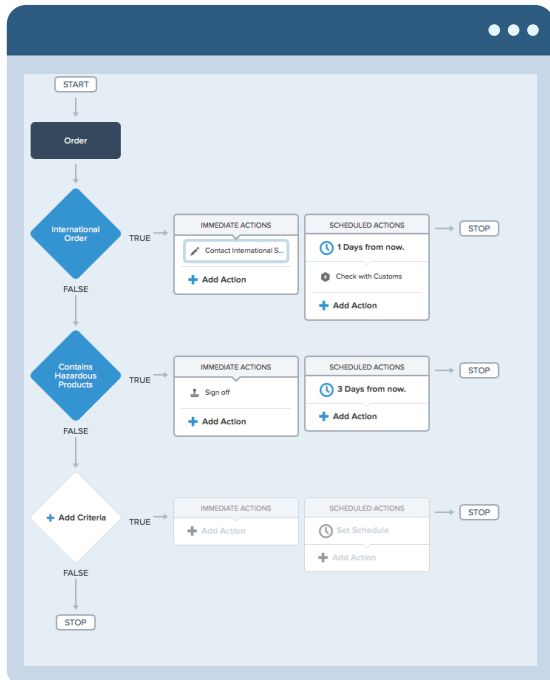
3つのアプローチのうち、もっとも進んでいるのがプロセス駆動型マイクロサービスです。ここでは、プロセス駆動型マイクロサービスの標準的な構成要素について説明します。

3.2.1 データオブジェクトの API 公開の自動化

前項でも紹介したこの機能では、特定のデータエンティティ（注文など）に対する CRUD 操作に特化した、専用のエンドポイントを提供します。ビジネスプロセスに関連するロジックや変換ロジックは含まれません。ビジネスユーザーは、スキーマビルダーを使用してこのサービスを作成できます。また、開発者はカスタムマイクロサービスの要領で自身で作成することもできます。

3.2.2 Lightning プロセスビルダーで作成されるビジュアルプロセス

App Cloud の [Lightning プロセスビルダー（英語）](#) は、ビジネスプロセスを設計するためのビジュアルツールを提供します。これらのプロセスには、データオブジェクトの作成や更新（自動化されたマイクロサービスによる処理が一般的）をトリガとするエントリポイントが含まれます。



独立した疎結合のマイクロサービスを開発するプロセス駆動型パターンは、拡張性の高さという点で最適なアプローチと言えます。

ビジネスユーザーは、プロセスビルダーを使用してプロセスの作成や変更を簡単に行えるほか、スキーマビルダーでデータオブジェクトマイクロサービスを作成することもできます。また、開発者はコードを使用してカスタムマイクロサービスを作成することも可能です。

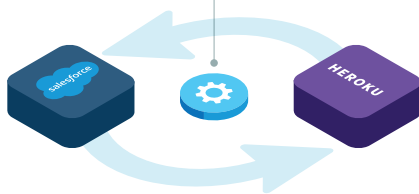
プロセス駆動型マイクロサービスでは、各種のサービスを疎結合させることで、データオブジェクトのアクセス方法や更新方法には関係なく、設計に沿ったプロセスを実行できます。

3.3 カスタムマイクロサービス

カスタムマイクロサービスを開発する必要性に迫られることは、少なくありません。カスタムマイクロサービスでは、前述のデータオブジェクトから情報を取得し、追加のロジックを実行したり、ほかのクラウドベースサービスを利用してデータを拡充したりできます。また、WS-* 規格ベースのマイクロサービスにありがちな弱点である、トランザクション制御とロールバックのサポートが必要となる各種の操作を実行することも可能です。App Cloud では、次の 2 つの手法でカスタムマイクロサービスのアーキテクチャを実現します。

3.3.1 任意の言語を使用し、Heroku でデプロイ

開発者は言語を自由に選択してカスタムマイクロサービスを記述し、Heroku を通じてデプロイできます。さらに、Heroku Connect で Salesforce とシームレスにデータを同期でき、マイクロサービスに連携ロジックを含める必要はありません。



その結果、それぞれ独立したマイクロサービスを疎結合して個別に保守できます。

```
#!/usr/bin/env ruby
require 'sinatra'
require './orderPrice'

oPrice = orderPrice.new

post '/price' do
  req = JSON.parse(request.body.read)

  args = req['params'].unshift(req['method'])
  result = oPrice.send *args
```

3.3.2 Apex で記述された、アノテーション付きのオブジェクト指向クラスとメソッドを使用

Apex は各種機能を完備した Java ライクな言語で、マルチテナント環境向けに設計されています。データバインドを効率化し、プログラム型の強力なアプリケーション構築機能を提供するほか、ロールとプロファイルベースのセキュリティやトランザクション制御などのプラットフォーム固有の機能をサポートします。開発者は Apex を使ってマイクロサービスを作成し、シンプルなアノテーションを追加して、RESTful なエンドポイントを定義できます。

```
@RestResource(urlMapping='/Account/**')
global with sharing class MyRestResource {

  @HttpGet
  global static Account doGet() {
    RestRequest req = RestContext.request;
    RestResponse res = RestContext.response;
    String accountId =
      req.requestURI.substring(req.requestURI.lastIndexOf('/')+1);
    Account result = [SELECT Id, Name, Phone, Website FROM
      Account WHERE Id = :accountId];
    return result;
  }

  @HttpPost
  global static String doPost(String name,
    String phone, String website) {
    Account account = new Account();
    account.Name = name;
    account.phone = phone;
    account.website = website;
    insert account;
    return account.Id;
  }
}
```

3.3.3 アクションベースのカスタムマイクロサービス

従来の SOA のアプローチは、主にアーキテクトと開発者のための手法でした。一方、App Cloud では、ビジネスユーザーでもコードを 1 行も書くことなくアクションベースのカスタムマイクロサービスを作成できます。



ユーザーが外出先でクイックアクションを実行するために設計されたこれらのサービスは、Salesforce1 モバイルアプリに自動で表示されます。また、App Cloud 上でアプリケーションを構築する開発者からもこれらのサービスにアクセスできます。



THE CUSTOMER SUCCESS PLATFORM

SALES SERVICE MARKETING COMMUNITY ANALYTICS APPS