

OutSystems におけるサポート外 DBMS の 利用方法について

OutSystems は内部 DB でデータを保持できるだけでなく、OutSystems の外部にある DB とも連携が可能です。主要な DBMS はサポートされていますが、サポート対象外の DBMS もあります。サポート対象外の DBMS を扱う場合、ODBC/JDBC コネクタを利用することが一般的です。

本資料ではサポート外の DBMS に焦点を当て、OutSystems との連携方法について説明いたします。

目次

1. サポート対象の DBMS	3
2. サポート外・外部 DB に接続するための拡張コード作成.....	3
3. サポート外・外部 DB 接続パターンについて	5
終わりに	7

1. サポート対象の DBMS

OutSystems でサポートされている DBMS は、以下の通りです。（2022 年 7 月時点）

- DB2 iSeries
- PostgreSQL and Aurora PostgreSQL
- MySQL
- Oracle
- Azure SQL and SQL Server

詳細なバージョンは、OutSystems サポート HP でご確認ください。

[サポート HP URL](#) :

上記以外の DBMS は、OutSystems のサポート対象外 DBMS（以下、サポート外・外部 DB）となるため、拡張コードを作成して連携する必要があります。

2. サポート外・外部 DB に接続するための拡張コード作成

前述の通り、OutSystems ではサポート外・外部 DB にアクセスするためには、Extension とよばれる拡張コードを作成して接続する必要があります。Extension は、OutSystems が提供している Integration Studio を通して C# で実装する外部プログラムです。

C# は Microsoft 社の Visual Studio で作成しますので、開発用の PC には OutSystems 以外に Microsoft 社の Visual Studio が事前にインストールされている必要があります。

1. Extension の定義

Integration Studio で Extension を定義します。定義するものは、以下となります。

- ・ Extension 名 : Service Studio のアプリケーション名
- ・ Action 名 : Service Studio の Action 名
Integration Studio を通して自動生成される C# コードの名前空間やメソッド名にも使用されます。
- ・ Action の Input/Output : Extension の Input/Output 変数を定義

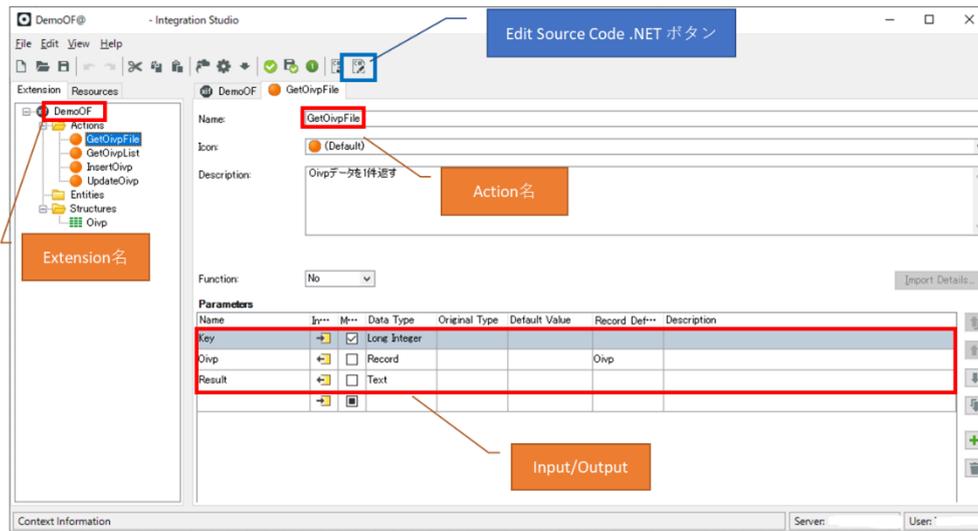


図 1.OutSystems の Integration Studio での設定

Integration Studio で Extension の定義が出来たら、Visual Studio でデータ操作のためのコードを C# で作成します。Integration Studio から「Edit Source Code .NET」を経由することで、OutSystems が C#コードを自動生成します。

2. C#でデータ操作を行うコードの作成

サポート外・外部 DB に接続するための ODBC/JDBC 接続に必要な接続文字列と、データ操作（取得・更新）のための SQL を含むコードを記述します。コーディングが終了したら、ビルドを行います。

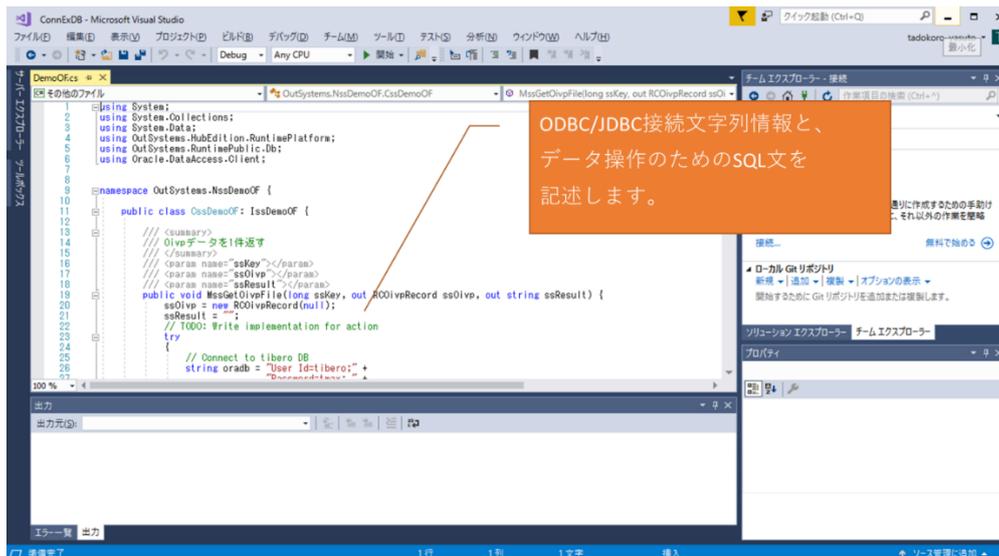


図 2. Visual Studio での C#コード

3. Integration Studio で Publish の実施

Visual Studio でソリューションのビルドが完了したら、Integration Studio に戻り Extension を Publish し、OutSystems の Service Studio で利用可能な状態にします。



図 3.Publish ボタン

4. 実装

Service Studio で作成した Extension を実装します。実装の仕方はサポート外・外部 DB だからと言って特別なことはなく、他の Action と同じとなります。以下が Service Studio での実装イメージになります。

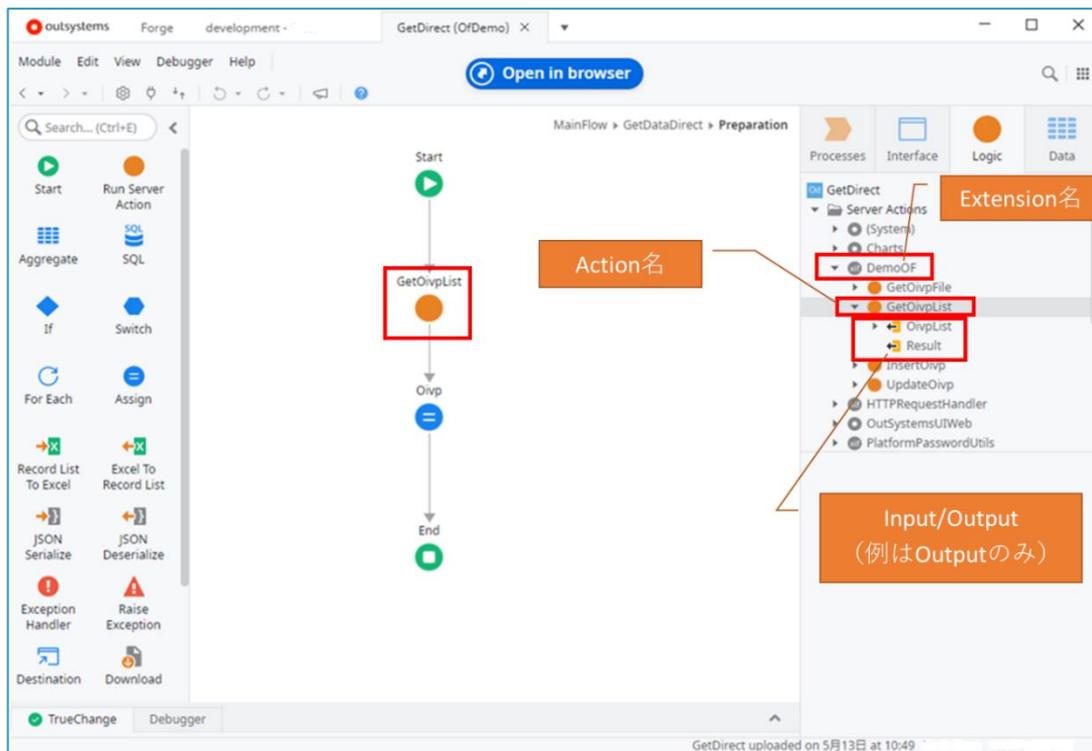


図 4. OutSystems の Service Studio での実装

3. サポート外・外部 DB 接続パターンについて

Extension を使用することでサポート外・外部 DB への接続はできますが、アプリケーションを構築するうえでデータ連携を行う方法とタイミングを考えてみます。

1つ目は、Extension を直接、画面・ロジックで使用するパターン、2つ目は、OutSystems のデータとして扱うことができる Entity を利用するパターンになります。それぞれの利用パターンについて解説いたします。

パターン 1

OutSystems で作成したアプリケーションから Extension を使用し、サポート外・外部 DB より直接データを操作する方法です。アプリケーションがデータを取得・更新するタイミングでサポート外・外部 DB に対してリアルタイムにアクセスします。構成イメージは以下のようになります。

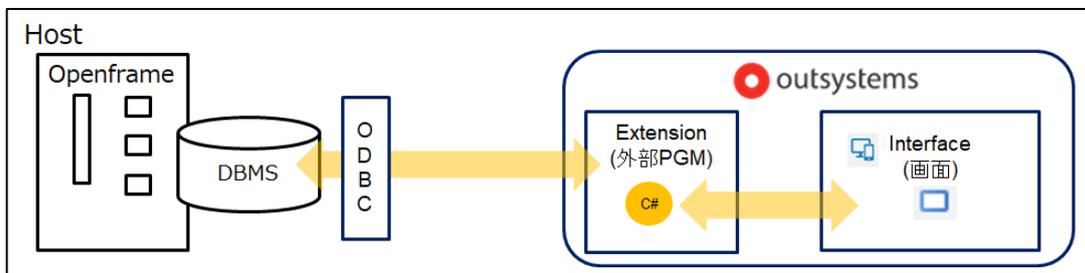


図 5. データ取得パターン 1 構成イメージ

この利用方法のメリットとしては、DBMS が他のシステムと DB を共有している場合、お互いのシステムでの更新タイムラグが発生しないことです。デメリットは、DB アクセスをすべて Extension 経由で行う必要があるため、OutSystems の機能を十分に引き出すことができません。

例えば、Entity を使用した画面自動生成（スキャフォールディング）、ブートストラップ、CRUD などの機能の使用ができません。また、データ抽出の条件や、更新条件などが画面機能ごとに異なる場合、その単位で Extension を作成する必要があります。

パターン 2

Extension でサポート外・外部 DB のデータを取得し、OutSystems 側のデータセットである Entity に保持する方法です。アプリケーションは Entity に対し操作を行い、サポート外・外部 DB との連携は非同期になります。構成イメージは以下となります。

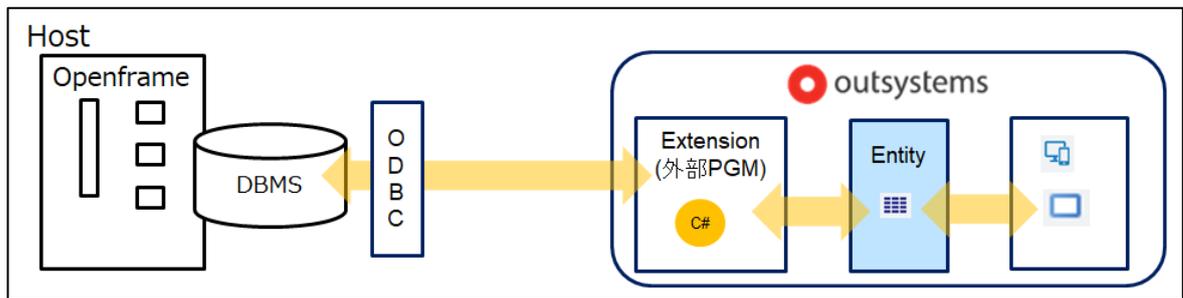


図 6. データ取得パターン 2 構成イメージ

このパターンのメリットは、パターン 1 では利用できない機能が利用でき、OutSystems のメリットを享受できることです。

デメリットは、OutSystems 側でローカルデータを保持することになるので、DBMS 側のオリジナルのデータとの差が発生し、その同期を取る必要がある点です。

また、取り扱うデータが大量である場合、システムを利用していない時間帯（夜間など）にバッチ処理を実行し、データ同期を行うなど、データの整合を考慮するための運用設計が必要になります。

終わりに

サポート外・外部 DB を OutSystems で取り扱う方法を説明いたしました。

制約事項はありますが、OutSystems のサポート外・外部 DB でも使用できることがお判りいただけたと思います。

tdi では、OutSystems の機能や技術について十分な知識を持った多くの技術者を有しており、資格保有者数は国内トップクラスです。ローコード開発が一般的に注目される以前（2016 年）から重ねた OutSystems 開発の実績をもとに、IT 戦略コンサルティングや OutSystems 導入から運用までをトータルサポートします。また、お客様に合わせた人材育成や内製化もご支援いたします。本件を含め、何かお困りごとがございましたらどうぞお気軽にお問合せください。



【ローコード開発基盤「OutSystems」】

<https://www.tdi.co.jp/outsystems/>

【お問い合わせ】

<https://tdi.smkgtg.jp/public/application/add/1095>

tdi 情報技術開発株式会社 営業本部

東京:〒163-1332 東京都新宿区西新宿六丁目5番1号 新宿アイランドタワー32階
03-5325-4811 (代表) FAX 03-5325-4812

中部:〒451-6027 愛知県名古屋市西区牛島町6番1号 名古屋ルーセントタワー27階
TEL 052-571-6871 (代表) FAX 052-571-3856

関西:〒530-0005 大阪府大阪市北区中之島二丁目2番7号 中之島セントラルタワー20階
TEL.06-6201-7739(代表) FAX.06-6201-7740

九州:〒812-0013 福岡県福岡市博多区博多駅東二丁目10番1号 福岡ビルS館7階
TEL.092-451-8218(代表) FAX.092-474-7379