

PART 3

ツール編

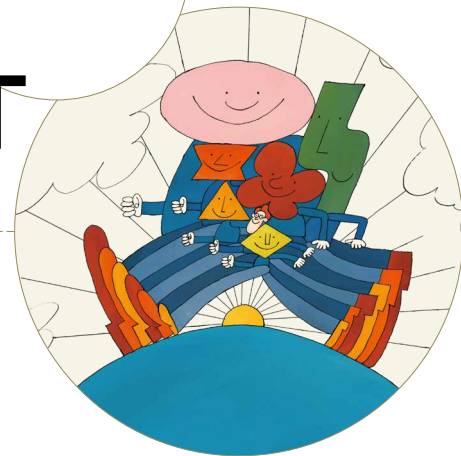
GoAnywhere MFT

株式会社ソルパック

APIをはじめシステム間の 多彩な連携を司る連携ツール

データ連携機能を司る「プロジェクト」を中心に
双方向のAPIを作成

Text= 栃内 勇紀 田中 遥 株式会社ソルパック



GoAnywhere MFT とは

GoAnywhere MFT（以下、GoAnywhere）とは、組織内のすべてのファイル転送を単一のインターフェースで管理するソリューションである。ネットワーク内にGoAnywhereを導入することで、サーバー間のファイル転送の自動化、個人間のファイル共有、コラボレーションを容易に実現する。

マルチプラットフォームに対応しており、IBM i への導入も可能である（本稿では、IBM i へのGoAnywhere導入を前提とするが、導入先プラットフォームによる機能や管理メニューに違いはない）。

GoAnywhereは、クライアント機能とサーバー機能の双方を備えている。クライアント機能はあらゆるシステムやサービスへ、GoAnywhereをクライアント側として接続し、マルチステップの連携処理を実行する（ファイル転送プロトコル、データベース接続、API呼び出しなど）。

一方のサーバー機能は、GoAnywhereがファイル転送プロトコルのサーバー側機能を提供する（FTPS、SFTP、HTTPS、AS2、API化など）。

GoAnywhereには、エンドユーザー自身で連携処理の定義作成やAPIを作成するための管理インターフェースが提供されている。

ここでは、IBM i 上に導入したGoAnywhere から他シ

ステム/サービスのREST APIを実行する方法、および他システム/サービスにREST APIを提供する方法の具体的なステップを紹介する。

IBM i からREST APIを実行する方法

まず一般的にREST APIが提供する情報を例に、IBM i からREST APIを実行する方法を説明しよう。

接続先側のAPI設定

一般にREST APIでは、サービスに応じて多様な認証方式が登場する。たとえばパスワード認証、APIトークン、セッション認証、OAuthなど。中でもAPIトークンのように、アプリケーション（この場合はRESTクライアント側となるGoAnywhere）に対して固有のトークンをサービス側で事前生成する必要がある場合は、事前準備としてREST APIガイドに沿ってサービス側の設定を行う。

接続元側の設定

サービスごとにAPIのガイドの公開方法は異なる。一般的にインターネット上に公開されているパターン、契約者に限定で公開しているパターン、エンドユーザー/パートナー向けなどさまざまだが、次のような構成で記載されていることが多い。

- ・共通仕様（認証方式を含む）
- ・REST API一覧

サービス側の設定も済み、ガイドも手に入れ、いざ開発となった場合に、IBM i上に導入されているGoAnywhereではどのように開始するかを以下に説明する。

リソース登録

APIガイドの「共通仕様」を見ると、多くの場合はホスト名、認証方式、リクエストヘッダのような情報が与えられる。たとえば、**図表1**のような情報が共通仕様としてガイドに示されている場合、GoAnywhereでは事前定義の接続設定「リソース」として登録する

「リソース」設定では、接続先となるサーバー、サービスのホスト名や認証情報を事前登録する。**図表1**のような情報が与えられた場合、GoAnywhereではリソースタイプ

図表 1 APIガイドの共通仕様の例

URL	https://sample.com/sample/rest/v1
データ形式	JSON (APIごとに異なる値を指定する場合もあり)
認証方式	Basic

図表 2 リソースタイプ「REST サーバ」の設定画面



図表 3 リソース設定のテスト機能使用イメージ



「RESTサーバ」として登録する（**図表2**）。

リソースの設定画面では、その場で接続可否を確認するためのテスト機能（**図表3**）が提供され、一度登録したリソースは一意の名称によって後述のプロジェクト（複数の連携処理からなる定義体）作成時に繰り返し呼び出せる（APIキーなど、その他のリクエストヘッダが指定されている場合は後述のプロジェクト内で設定する）。

プロジェクト作成

「共通仕様」部分をリソースとして登録後は、「API一覧」で指示される内容に沿って連携処理をデザインする。一般的に「API一覧」では、サービスに対するリソースの追加、取得、更新に必要なリクエストおよび期待されるレスポンスを記載することが多い。

たとえば、**図表4**のようなAPI情報が提供されている場合、GoAnywhereでは「プロジェクト」内で定義する（ここでは**図表1**で与えられた共通仕様と関連のあるAPIとする）。

「プロジェクト」とは、複数の連携処理（ファイル連携、DB連携、API連携）の組み合わせからなる定義体の基本単位である。プロジェクトでは直観的に操作できるデザイナー画面（**図表5**）が提供され、画面左側「コンポーネントライブラリ」、画面中央「プロジェクトの概要」、画面右側の「作業ウインドウ」により構成される。

コンポーネントライブラリには、プロジェクトに含められる連携処理やローカルファイル処理などがリストされており（GoAnywhereでは1つ1つの処理を「タスク」と呼ぶ）、この中から必要な処理を選択して「プロジェクトの概要」に追加し、追加した処理の設定を「作業ウインドウ」で行う、といった操作で定義をデザインできる。

図表4のAPIが与えられた場合、具体的には**図表5**のサンプル画面で示しているような「REST Put」タスクと「Read JSON」タスクを組み合わせる。「REST Put」タスクでリクエストを送信し、「Read JSON」タスクでレスポンスのデータを読み込む。

こうした操作により、IBM i上に導入したGoAnywhereからREST APIへの連携が実現する。5250画面上の操作は不要で、GoAnywhereの管理画面上のみで設定できるのがポイントである。

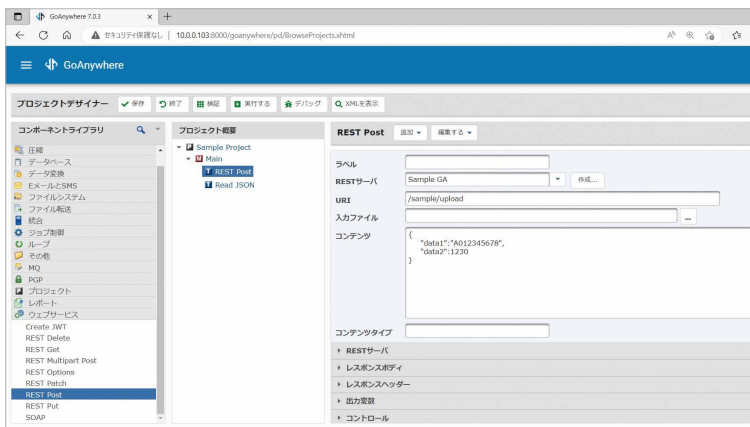
ただし実運用を考慮すると、APIへの連携のみで定義は完結しない。リソースを更新するAPIであれば、Db2 for iからデータを取得する処理が必要であり、リソースを取得するAPIであれば、取得したデータをDb2 for iに連携する処理も必要となる。

Db2 for iと連携するパターンは、大きく2つある（**図表6**）。RPGと連携した処理のパターン、およびDb2 for iと

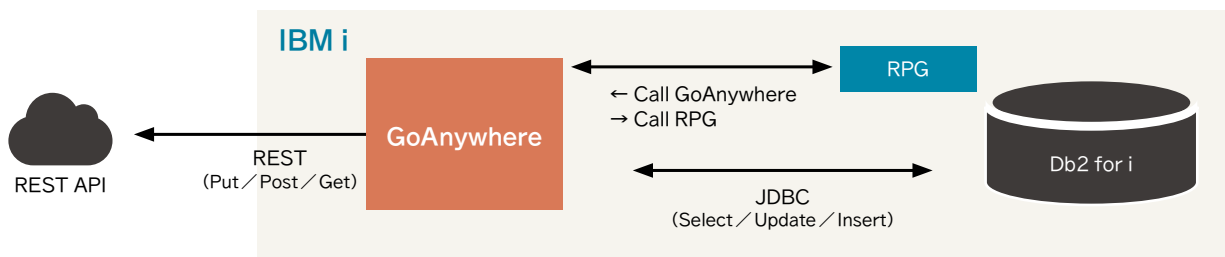
図表 4 リソースを更新するAPIの例

リクエスト	
HTTP メソッド	PUT
URI	/sample/update
リクエストボディ	data1 data2
レスポンス	
ステータスコード	200
データ形式	JSON
レスポンスデータ	output_data1 output_data2

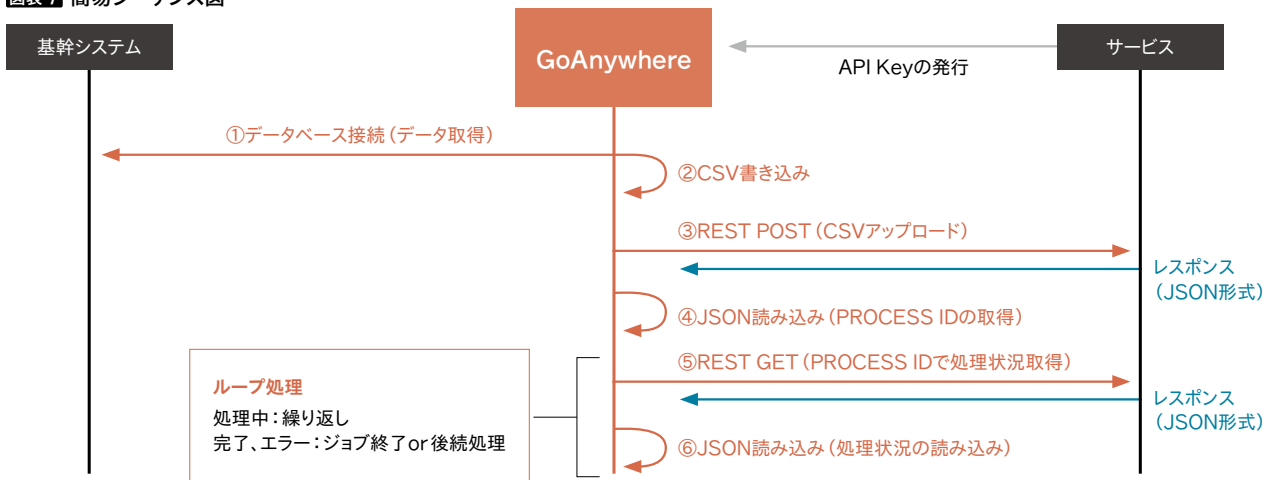
図表 5 プロジェクトデザイナー画面



図表 6 Db2 for i とREST APIの連携パターン



図表 7 簡易シーケンス図



直接JDBCで接続するパターンである。GoAnywhereは
いずれのパターンでも問題なく活用できるが、実装後の運
用を考えると、Db2 for iと直接接続するパターンのほうが
管理はシンプルになる。

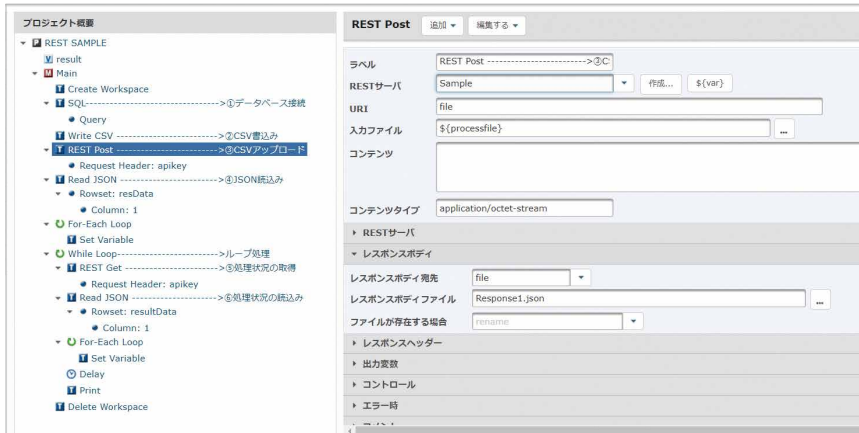
また、API連携そのものが複雑なケースも想定せねばな
らない。本稿では極めて簡単な例を取り上げているが、実
際にはリクエストの送信、もしくは必要な情報を得るまで

に複数のステップを要する場合が多い。認証に2つ以上の
ステップを要する場合もある。

実際に起こり得る連携処理として、Db2 for iとの連携、
および複数のAPI連携が必要な例を図表7に示した。

処理の概要としては、Db2 for iのデータをCSVファ
イルに変換し、REST APIを実行してCSVファイルをアップ
ロードしている。アップロードしたCSVファイルは、サー

図表 8 Db2 for iとAPI連携を組み合わせたプロジェクトの例



実行／自動化

プロジェクトをデザインしたあとは、実行を自動化する方法を検討する。GoAnywhereは、スケジューラー、モニター（フォルダ監視機能）、コマンド実行（IBM i、Windows、AIX、UNIX、Linux向け）、REST／SOAPなど、自動化のオプションを複数提供している。

またGoAnywhere 上ではジョブキューも提供しており、優先度に応じて複数のジョブを柔軟に管理できる。

IBM i 上の実装では、GoAnywhere 実行コマンドをCLプログラム／RPG

ビス側でバッチ処理されるが、処理状況を確認するAPIを繰り返し実行することで、正常に処理が終了しているかを検知している。

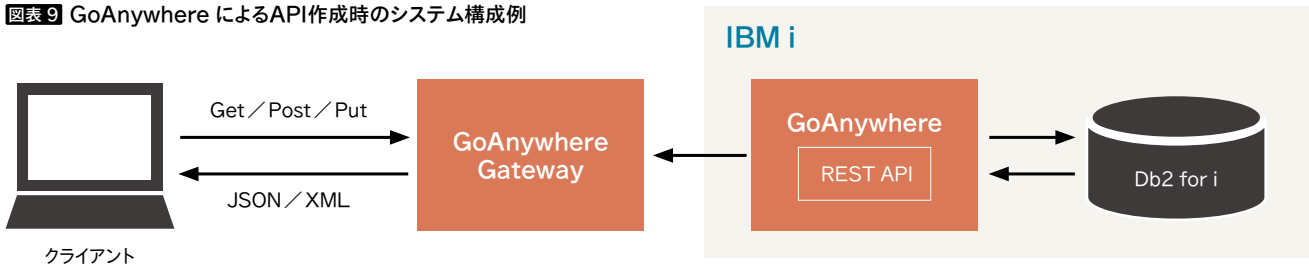
この処理に従って作成したプロジェクトが、図表8である。定義の複雑性はあるものの、1つのプロジェクトのみで、Db2 for iとAPIの連携を完結できる。

に組み込んだ自動化、もしくはGoAnywhere のスケジューラーをベースとした自動化が多く利用される。

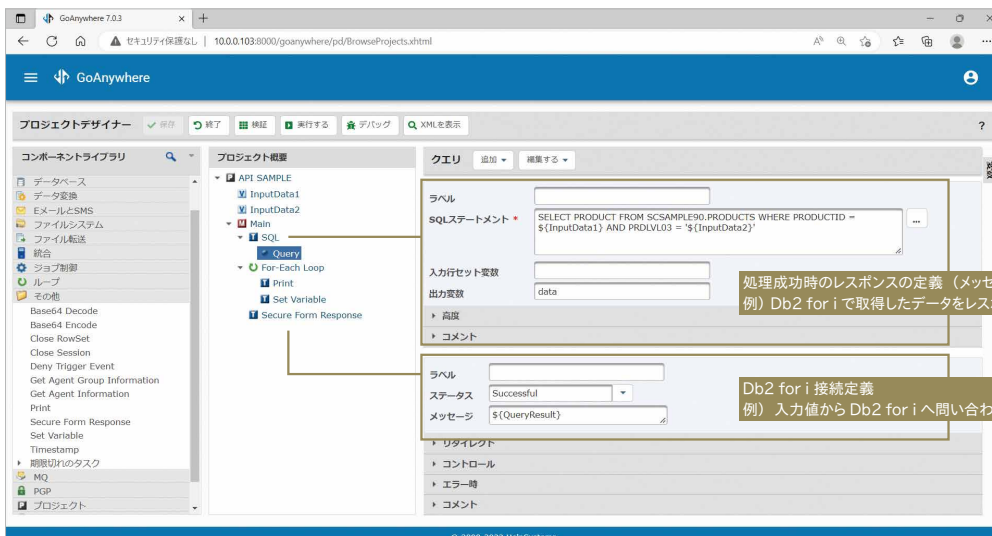
他のシステム／サービスから IBM i を操作するためのAPI作成方法

次に、他のシステムやサービスからIBM i 上のリソース

図表 9 GoAnywhere によるAPI作成時のシステム構成例



図表 10 API化を想定したプロジェクトのサンプル画面



処理成功時のレスポンスの定義（メッセージに任意のデータを指定できる）
例） Db2 for i で取得したデータをレスポンスとして設定

Db2 for i 接続定義
例） 入力値から Db2 for i へ問い合わせを行う処理

を操作するためのAPIを作成する方法を紹介する。

システム構成

まずシステム構成では (APIの公開範囲にもよるが)、IBM i に対する通信が要求されるため、セキュリティ面を考慮する必要がある。GoAnywhere では、図表9に示す構成を提案している。

ここではIBM i に導入したGoAnywhere とGoAnywhere Gatewayを組み合わせている。GoAnywhere Gateway は、GoAnywhere とインターネット間のリバースプロキシとして稼働し、DMZへの導入を想定している。GoAnywhere Gateway からGoAnywhere へは通信を行うことなく、外部からAPIへのアクセスを可能とする。

APIの作成

システム構成が決まれば、簡単にAPIを作成できる。前述したプロジェクトを用いるが、このプロジェクトではRESTでの呼び出しが可能なので、レスポンスなどの拡張設定を行うだけでAPIを作成できる。

●プロジェクトの作成

他システムからRESTで呼び出された場合、プロジェクトがDb2 for iなどと対話する。プロジェクトの作成方法は前述したとおりで、API化を想定した場合、クライアントからの入力値とクライアントへのレスポンスを意識した定義を行う。

入力値はDb2 for iへの問い合わせのキー項目や、データ処理の入力値に使用したり、レスポンスはプロジェクト内の一連の処理で出力されたデータを用いるのが一般的である。

簡易な例として、2つの入力値からDb2 for iへ問い合わせを行い、取得したデータをレスポンスに設定したプロジェクトのサンプル画面が図表10である。入力値は変数として定義し、レスポンスは専用の設定項目を使用する。

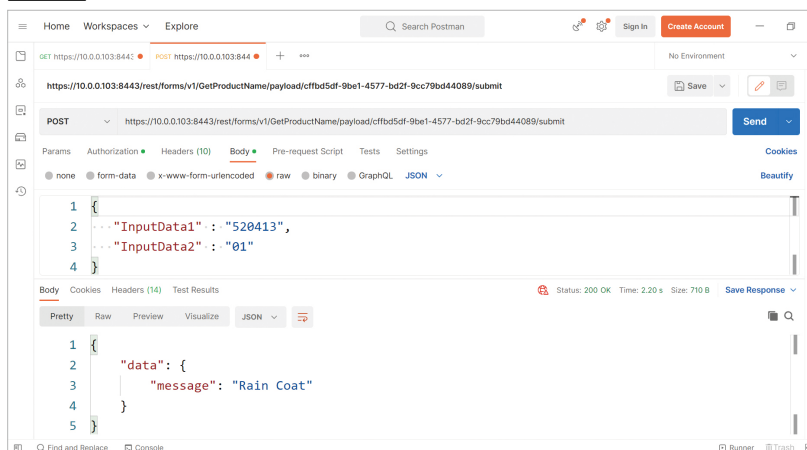
●プロジェクトのAPI有効化

プロジェクトを作成したら、API化のための設定を行う。プロジェクトのAPI化専用の管理メニュー上で、入力値 (API実行時のリクエストボディのパラメータとなる) や使用ユーザーなどを設定する (図表11)。REST APIの有効化もチェックボックスだけで設定でき、有効化後はAPIを

図表 11 プロジェクトをAPI化する設定画面サンプル



図表 12 Postman によるAPI実行イメージ



使用するためのヘルプ画面などもある。

以上が、GoAnywhere によってIBM i 上のリソースを操作するためのAPIを提供する手順となる。実際にサンプルで使用したAPI (2つの入力値から製品マスターを参照して製品名を取得) を実行したのが、図表12である。

本稿では、IBM i 上に導入したGoAnywhere によってREST API を実行する手順、またそれとは逆に、IBM i のリソースを外部から操作するAPI作成の手順を示した。

どちらの場合もデータ連携機能を司る「プロジェクト」を中心に、Db2 fo i と対話するシンプルな例で説明したが、より複雑な要件にも対応可能である。📌