APIエコノミーの価値とそれを実現する技術

FinTechとは、金融機関やそれ以外の企業がITを駆使して提供する新しい金融サービスや、それを利用した異業種連携のことを指します。昨今、FinTech市場の急速な拡大に後押しされ、複数の企業がAPIを公開することでAPIエコノミーを形成することが注目されています。本稿では、APIエコノミーの形成により創出される新たな価値と、それを実現するためのシステム技術について解説します。

1. APIエコノミーとは何か？

今、APIエコノミーが注目されています。プログラムの機能を外部から呼び出して利用できるようにする「API（Application Programming Interface）」という技術自体は以前から利用され、開発の重複を避け既存サービスの再利用による開発コスト削減のための努力も行われてきました。

では今、APIがこれほど注目されるようになった理由は何なのでしょうか。それは今日起きているAPI公開が、単に社内の別プログラムからの呼び出しだけではなく、社外のプログラム、それも主にモバイル・アプリから利用されることを想定しているからです。急増するモバイル・ユーザーの要望に応えるために、アプリ開発者はユーザー視点に立った便利な機能を素早く提供する必要に迫られています。そのためには企業が持つサービスやデータを活用し、組み合わせて、業界を飛び越えたサービスを提供することが必要となったのです。

企業が自社の持つデータやサービスにAPIを生（は）やし、外部のアプリからWeb経由で簡単にアクセスできるようになると、自社では到底思いつかなかった新たなビジネスを外部のアプリ開発者が作り出してくれる可能性が出てきます。そしてそのアプリを利用するユーザーは、アプリが提供する機能やキャンペーンによってAPI提供企業のサービスを利用するようになり、間接送客が発生するようになります（**図1**）。

2. API公開がもたらす効果

モバイル・アプリを自社開発している企業は、API公開によってアプリ開発をアウトソースすることが可能に

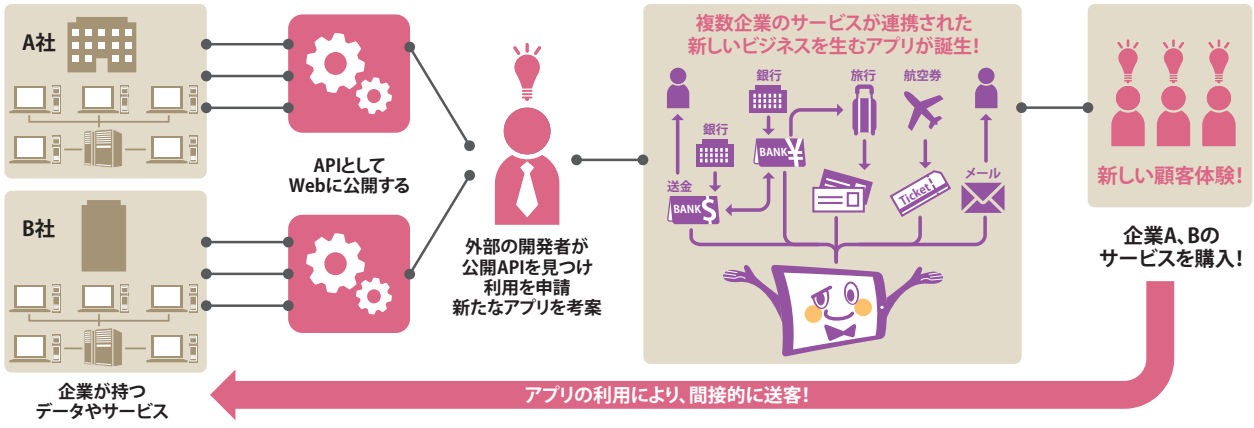

図1. APIエコノミーのバリューチェーン

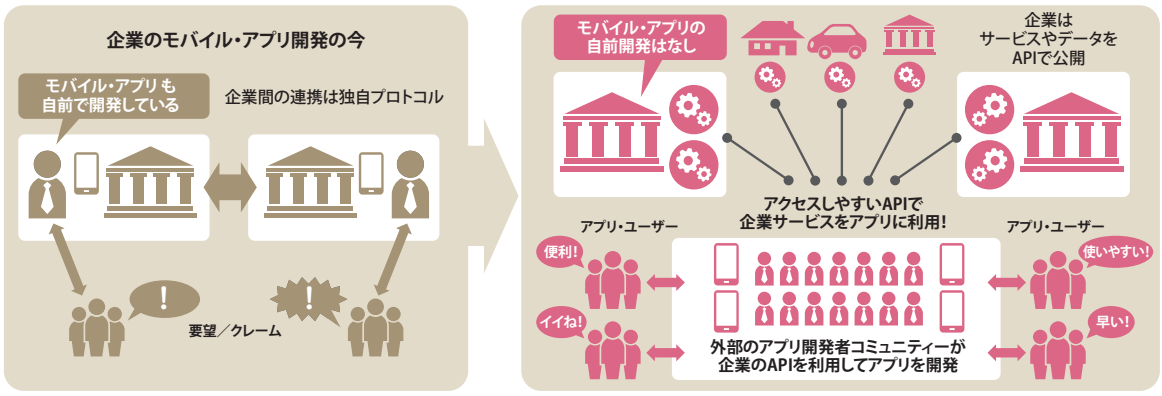

図2. モバイル・アプリ開発のアウトソース

なります。エスカレートするモバイル・ユーザーの要望にタイムリーに応え続けるには開発コストが発生しますが、APIによって自社のサービスやデータを外部から安全に活用できるようにすれば、外部のアプリ開発者がその企業の価値の高いサービスを利用して効率的にアプリを開発できるようになり、ユーザーは高機能で便利なアプリを手に入れることができます。つまりAPIエコノミーは、企業（API提供者）とアプリ開発者（API利用者）とアプリ・ユーザーの三者によるシェアリング・エコノミーでもあります（**図2**）。

企業のAPI公開が進むと、FinTechアプリも金融業界を超えて成長することができます。口座アグリゲーションや家計簿などの個人財務管理機能を提供するPFM（Personal Financial Management）アプリはユーザーの貯蓄状況や消費活動の情報を蓄積しています。銀行はこの情報を活用して最適なタイミングでそのユーザーだけの特別なキャンペーンを提供することができます。また、FinTechアプリが旅行会社の旅行商品検索APIを利用し、ユーザーの好みにあったツアーを探し出し、申し込みAPIを呼び出してすぐに予約や購入を完了することもできます。ユーザーは自分で複数の旅行アプリを使ってツアーを探す手間が省け、お得なツアーをいち早く確保でき、金融機関にとっては送金サービスの利用が増える可能性があります。

多くの人は複数の金融機関に口座を持っており、金融機関が提供する個別のアプリやWebサイトにいちいちログインして残高を見るのを面倒だと感じています。PFMアプリはそれらをまとめて表示してくれるため、一度使い出したユーザーはその便利さを手放すことが難しくなります。ユーザーは頻繁にPFMアプリを起動しており、銀行からすると、ユーザーに魅力的な個別キャンペーンや新商品情報を確実に届ける格好の手段となります。

しかしながら現在はまだAPIを公開していない金融機関が多く、PFMアプリはユーザーのユーザーIDとパスワードを預かってスクリーン・スクレーピングをせざるを得ません。これは金融機関のWebサイトに負荷をかけるとともに、タイムリーなキャンペーン提供もままなりません。ユーザーも自分のユーザーIDとパスワー

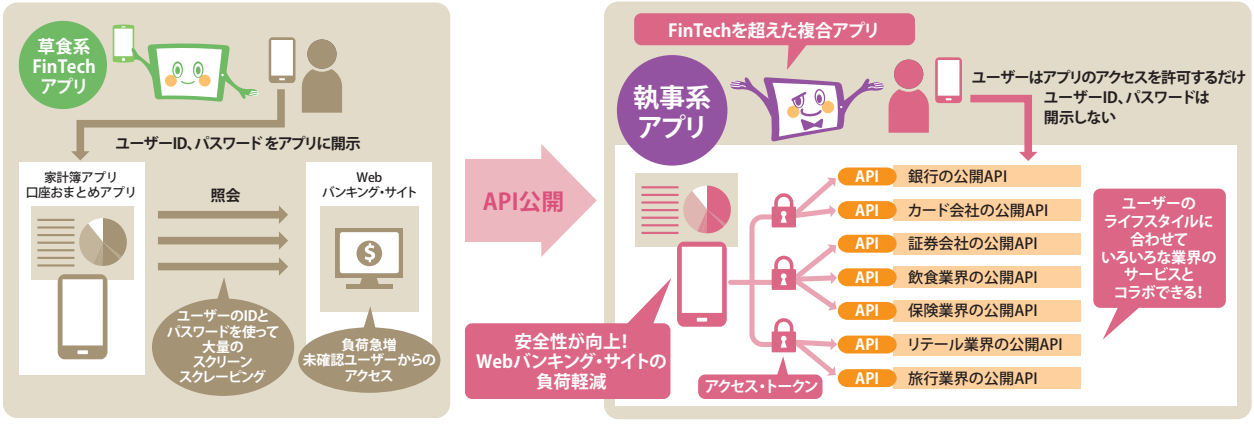

図3. FinTechアプリが業界を超える

ドをPFMアプリに登録することに抵抗を感じています。APIアクセスになれば、金融機関はアプリに対してAPIアクセスを許可するかどうかをコントロールし、そのアプリのユーザーを認証し、APIゲートウェイがその情報に基づいてAPIアクセスを一元管理することにより、アプリがユーザーIDやパスワードを預かることなく金融機関のサービスを利用できるようになります（**図3**）。

外部のアプリ開発者は、APIを公開している企業に対してシェアリング・エコノミーに前向きで革新的な文化を持っている印象を持つため、企業のブランド・イメージ向上につながります。企業がAPIを公開する場合は、適切に管理してその品質を保証し、アプリ開発者がそのAPIを発見しやすいようにしなければ、せっかく公開したAPIが利用されずビジネスにつながりません。スクレーピングでは誰がどの情報を取得したのか分かりませんが、APIアクセス状況をモニターすることができれば、どのアプリのどのユーザーがどのサービスやデータにアクセスしたのかを完全に把握できるようになり、企業データの管理強化につながります。また、各APIの利用状況を分析し、新たな商品としてのAPIをデザインすることも可能になります。API公開に必須となるこれらの管理機能は、「IBM API Connect」によって包括的に提供されています（**図4**）。

3. APIの技術面での概説

この章では、APIの技術について簡単に説明します。

現在、機器同士が連携するための仕組みとして、Web技術を応用してどんな機器でも連携できるようなシンプルな仕様とした、Representational State Transfer Application Programming Interface（REST API）が一般的となっており、最近では略してAPIと記載されることが多くなっています。REST APIでは、アクセス先の資源をUniform Resource Identifier（URI）で表現し、アクセス先の資源に対してHTTPのメソッドで操作要求すると、資源に対する操作結果を応答します。要求と応答形式は、項目名と値を列挙して表現するJavaScript Object Notation（JSON）[1]形式を利用するのが一般的です。また、APIの要求と応答の仕様の記述方法は、Swagger CommunityからOpen API Initiativeに寄贈されたOpenAPI Specification[2]で標準化されています。

4. API公開のために必要なシステム・コンポーネント

この章では、API公開を実現するために必要なシステム・コンポーネントについて説明します。

APIを利用して、既存の業務システム機能をインターネットに公開する際には、既存インターフェースを利用して業務システム機能をAPIに変換するためのフロント・システム（**図5①**）、API利用時の認証を行うための認証システム（**図5②**）、インターネット経由でAPIを安全に公開するために認証連携、アクセス・流量制御などを実施するためのAPIゲートウェイとそれらの構成管理・監視するためのAPI管理システム（**図5③**）が必要となります。

フロント・システムおよび認証システムは、既にインターネットに公開しているシステムを保有している場合は、そのシステムをAPI公開でも流用できる可能性が高

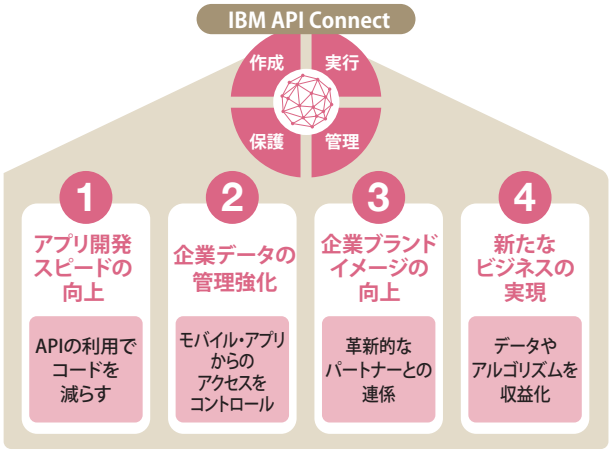

図4. API公開の価値とIBM API Connect

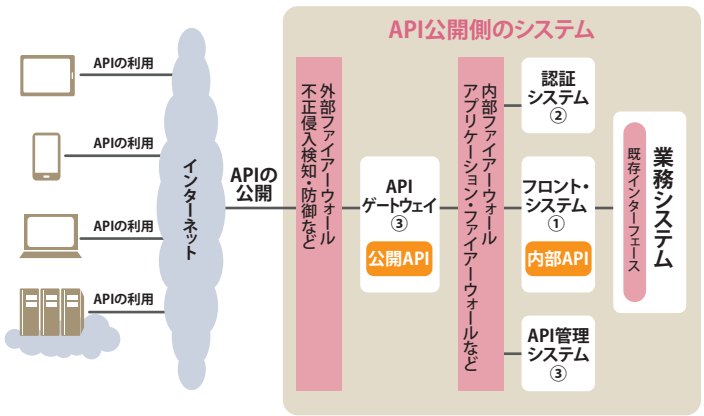

図5. 既存の業務システム機能を公開するシステム・イメージ

いと考えられます。

5. FinTech共通API

各社がAPI仕様を独自に定義してAPIを公開した場合、同じ業界内でも各社間でAPI仕様が異なってしまいます。各社間でAPI仕様が異なった場合、FinTech企業などAPIを利用する側は、同じアプリケーションを実装する場合でも各社のAPIごとに異なる実装を強いられます。

そこで、日本IBMでは、金融業界における昨今のFinTech市場の急速な拡大に対応して、API仕様の標準を定義し「FinTech共通APIアセット」(**図6**)[3]として提供しています。

FinTech共通APIアセットは、インターネット・バンキングやモバイル・アプリも対象としてビジネス・シナリオを標準化しているBIAN(Banking Industry Architecture Network)[4]をもとに、国内の銀行業務で必要なデータ項目も加味して標準仕様を定義しています。FinTech共通APIは、OpenAPI Specificationで定義したAPI仕様のほか、APIのテスト利用のためのチャレンジスタブ、サンプル・アプリ、他のAPIをFinTech共通APIに変換して公開するためのマッピング機能を提供しています。現在FinTech共通APIは、銀行業務、クレジットカード業務に対応しており、今後、他の業務にも順次対応していく予定です。

6. API公開における セキュリティー上のリスクと対策

APIを公開し、FinTech企業などの外部企業のサーバー・アプリからAPIを利用するといったユースケースを想定した場合、**図7**のようなセキュリティー上のリスクがあります。

APIを利用する際に利用者のユーザーIDとパスワードで認証する場合、API利用のためのユーザーIDとパスワードは外部企業が提供するモバイル・アプリやサーバーを経由することになります。そのため、マルウェア対策や外部企業側でのセキュリティー対策・運用が確実に実施されていないケースでは、利用者のユーザーIDとパスワードを漏えいさせてしまうリスクがあります(**図7①**)。

このリスクへの対策としては、OAuth 2.0[5]の認可フレームワークを利用して、API利用時の認証は外部企業が提供するモバイル・アプリやサーバーを経由せず、マルウェア対策や二要素認証などの高度な認証機能を提供可能なAPI公開側の認証システムで実施し、APIの利用を認可することを示すコード情報だけを外部企業のモバイル・アプリやサーバーに持ち回ってAPIを利用するといったやり方が一般的です(OAuth 2.0によるAPI利用の流れについては後述します)。

OAuth 2.0を利用した場合でも、API利用時に取得したユーザーに関するデータを外部企業のサーバーに保存することが可能です。外部企業側でセキュリティー対策・運用が確実に実施されていない場合は、前述と同様に、ユーザーに関するデータを漏えいさせてしまうリスクがあります(**図7②**)。このリスクへの対策としてAPI公開側の機能で情報漏えい自体を防止するのは困難なため、接続元の外部企業がセキュリティー対策・運用を確実に実施しているかを審査し、審査基準を満たした外部企業だけがAPIを利用できるようにアクセス制限する機能を設

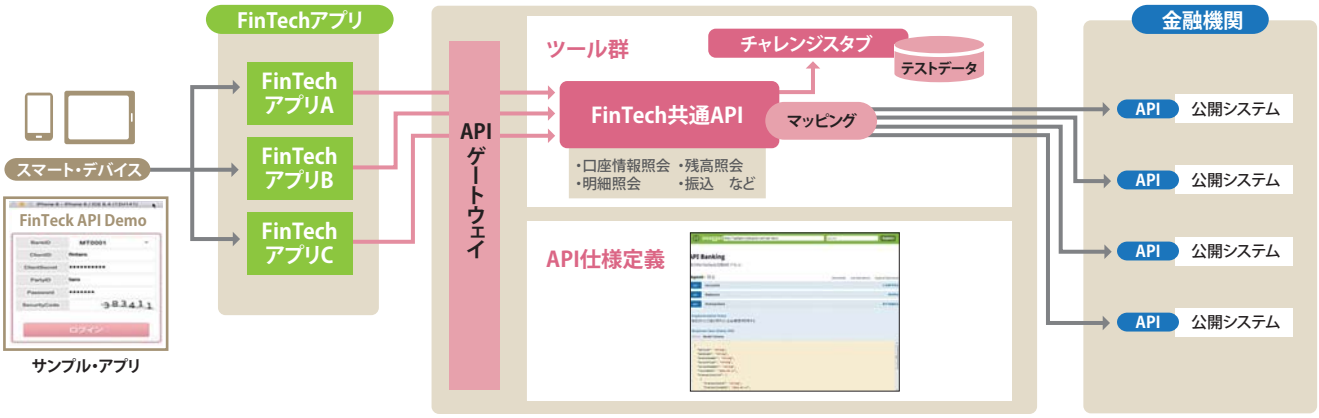

図6. FinTech共通APIアセットの概要図

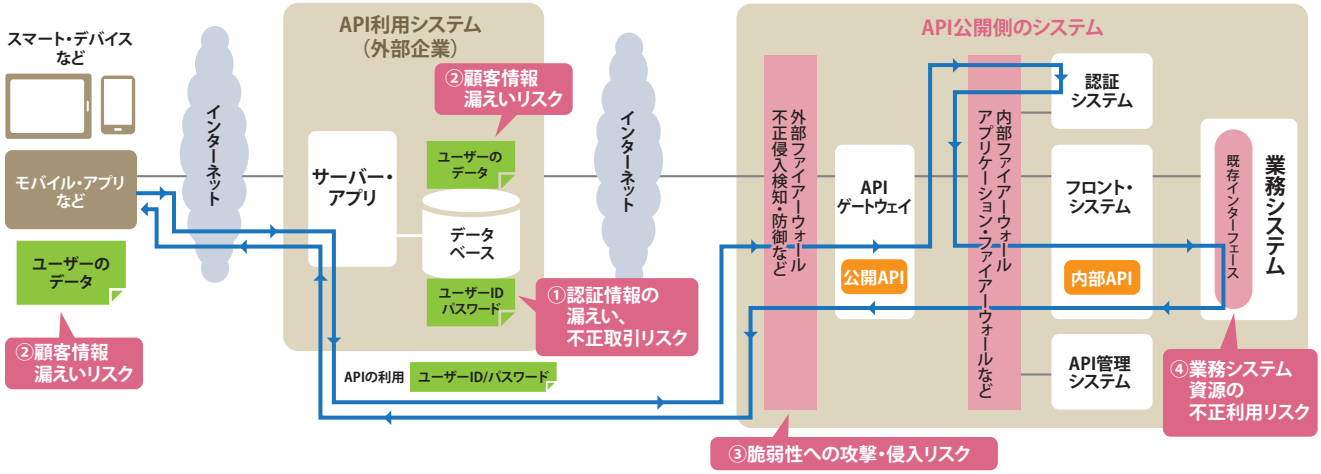

図7. API公開におけるセキュリティー上のリスク

けて運用していくほかありません。

また当然のことながら、API公開側のシステムにおいても、インターネットに公開するシステムと同様に、ファイアーウォールや不正侵入検知・防止などの基盤レベルのセキュリティー対策・運用（**図7③**）や、SQLインジェクション対策やAPIの入力パラメーターではなく認証したユーザーに紐付けたシステム資源へのアクセス制限などのアプリケーション・レベルのセキュリティー対策・運用（**図7④**）を確実に実施する必要があります。

7. OAuth 2.0によるAPI利用の流れ

前述と同様に、FinTech企業などの外部企業のサーバー・アプリからAPIを利用するというユースケースを想定した場合の、OAuth 2.0によるAPI利用の流れを**図8**に示します。

(1) 外部企業のサーバー・アプリからリダイレクトされる形で、API公開側のシステムで利用者の認証を行います。この処理は後述するアクセス・トークンが既に発行されている場合はスキップされます。認証された後、利用者に対してAPIの利用を外部企業に認可してもよいか確認が求められます。許可した場合は、APIの利用を認可するということを示す認可コードが発行されます（**図8①**）。

(2) 発行された認可コードを外部企業のサーバー・アプリに送信します。サーバー・アプリでは受信した認可コードをAPIゲートウェイに送信し、API利用を一定期間許可されていることを示すアクセス・トークンを取得します（**図8②③**）。なお、サーバー・アプリからAPIゲートウェイにアクセスする際には、利用者であればユーザーIDとパスワードに該当するようなサーバー・アプリ自体のAPIキー（ClientID/ClientSecret）によりサーバー・アプリを認証します。APIキーはAPI公開側のシステムで事前に発行します。

(3) 外部企業のサーバー・アプリからAPIゲートウェイにアクセス・トークンを送信しAPIを利用します（**図8④**）。アクセス・トークンは有効期限内であれば、再度（1）の流れから実施し直す必要はなく、アクセス・トークンを送信するだけでAPIを利用できます。そのため、アクセス・トークンはAPIの特性に応じて適切な有効期限を設定する必要があります。また、アクセス・トークンの有効期限が切れた場合にはアクセス・トークンを更新するときだけに利用するリフレッシュ・トークンも発行することができます。

なお、OAuth 2.0では明確に規定されていませんが、セキュリティーをより強化する場合は、サーバー・アプリのなりすまし防止のためにAPIキーだけでなく、証明書やソースIPアドレスなどによる追加の認証要否、利用者のユーザーIDとパスワード漏えい時などのアクセス・トークンの失効機能の要否を検討する必要があります。

8. API公開のセキュリティー対策における IBM API Connect / IBM DataPower Gatewayでの対応

前述したIBM API Connectは**図7**のAPI管理システム

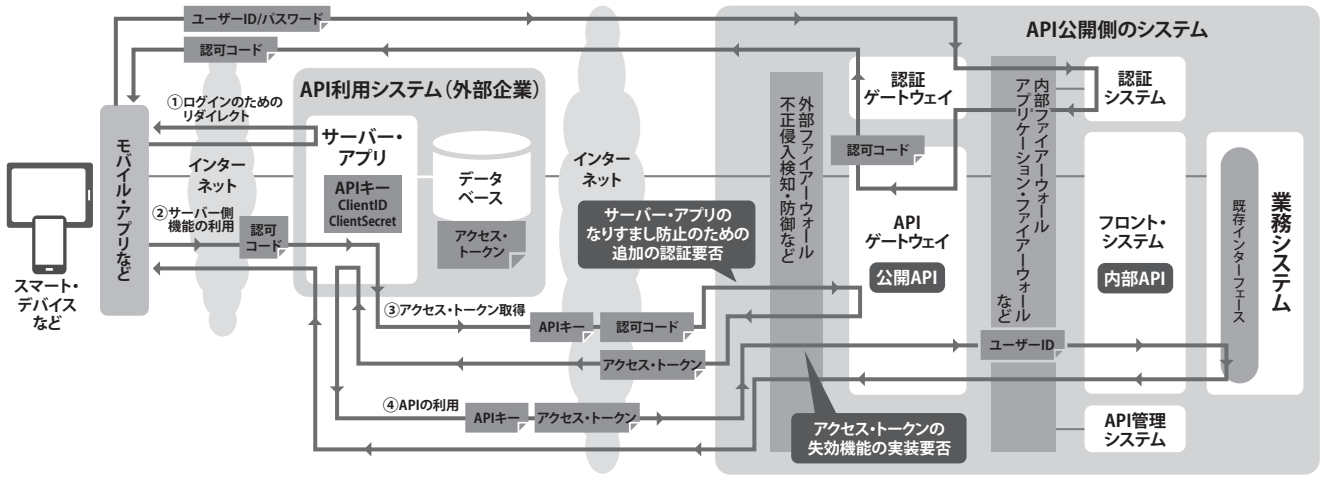

図8. OAuth 2.0によるAPI利用の流れ

に、また「IBM DataPower Gateway」はAPIゲートウェイとして位置付けられます。これらの製品により、前述したセキュリティー対策に関して、ネットワークレベル、アプリケーション自体で実装すべき要件を除き、すべての要件に対応することが可能です（**表1**）。

9. おわりに

本稿では、APIエコノミーの形成により創出される新たな価値と、それを実現するためのシステム技術や考慮点、製品での対応について解説しました。FinTechに関わらず、今後API公開を検討されている方々の一助となれば幸いです。

表1. セキュリティーに関する要件と
IBM API Connect / IBM DataPower Gatewayでの対応

関連するリスク		セキュリティーに関する要件	API Connect / DataPower Gatewayでの対応
1	図7 ①	認証ゲートウェイと連携し、OAuthの認可コードを発行する	API Connectではいくつかの認証連携パターンに対応可能
2	図7 ①	OAuthに従い、アクセス・トークン/リフレッシュ・トークンを発行し、トークンによるアクセス許可を行う	API Connectの標準機能で対応可能（実際にはDataPower Gateway上に構成が配布され実行される）
3	図7 ①	APIの特性に応じたトークンの有効期限を設定できる	APIごとにトークンの有効期限を設定可能
4	図7 ①	発行したトークンの失効ができる	外部のWebアプリケーションと連携することで、利用者単位、トークン単位でのトークンの失効が可能
5	図7 ②	外部アプリケーションに対して、APIキー（ClientID/ClientSecret）を発行し、API利用時にAPIキーをもとに外部アプリケーションの認証を行う	API Connectの標準機能で対応可能（実際にはDataPower Gateway上に構成が配布され実行される）
6	図7 ②	外部アプリケーションに対して、APIキーに加え、追加の認証を行う	DataPower Gatewayに追加の構成を行うことで、証明書やソースIPアドレスに対する追加の認証が実現可能
7	図7 ②	外部アプリケーションのアクセス許可の取り消しを行う	API Connectの標準の管理機能で対応可能
8	図7 ③	外部アプリケーションに対して流量制御を行う	外部アプリケーションとAPIの組み合わせ単位での流量制御を設定可能
9	図7 ③④	API利用時のリクエストデータによる攻撃を防止する	DataPower Gatewayに追加の構成を行うことで、データサイズやSQLインジェクションなどのリクエストデータに対する対策が可能

[参考文献]
[1] RFC7159, The JavaScript Object Notation (JSON) Data Interchange Format, https://tools.ietf.org/html/rfc7159
[2] OpenAPI Specification, https://www.openapis.org/specification/repo
[3] 日本IBM：「FinTech共通API」を提供　利用者の利便性向上、セキュリティー強化、開発生産性向上を支援，http://www-03.ibm.com/press/jp/ja/pressrelease/49204.wss
[4] BIAN, https://bian.org/
[5] RFC6749, The OAuth 2.0 Authorization Framework, https://tools.ietf.org/html/rfc6749

日本アイ・ビー・エム株式会社
グローバル・ビジネス・サービス事業本部
銀行・FMソリューション・デリバリー
シニア・アーキテクト

早川 勝
Masaru Hayakawa

1995年に入社後、テクニカルサービス部門に配属され、2003年にメガバンク担当サービス部門に異動。現在は、同メガバンク担当サービス部門のテクニカルリーダー、リードアーキテクトとして、プロジェクト全体をリードしつつ、FinTech、ブロックチェーンなどの最新技術については、金融サービス部門、日本IBM全体に対して技術推進を実施している。

日本アイ・ビー・エム株式会社
クラウド・テクニカル・サービス
エグゼクティブ・アーキテクト

早川 ゆき
Yuki Hayakawa

1987年入社。e-Commerceスペシャリスト、IBM Asia Pacific WebSphereテクニカルセールス・リーダーなどを経て、メガバンク担当ソフトウェア・アーキテクトとしてIBMソフトウェア開発ラボと連携しソフトウェア製品の品質改善活動などを推進。現在はIBMクラウドのソフトウェア・アーキテクトとして、多様な業界のビジネス・リーダーにお会いし、企業のAPI公開を推進している。