

# デジタル共創を支える システムエンジニアリング

## System Engineering Supporting Digital Co-creation

● 上原 忠弘

● 松尾 昭彦

---

### あらまし

あらゆるものがデジタル化され、それらを組み合わせて新たなビジネスを生み出すデジタル共創の世界が広がりつつある。企業がこの世界で生き残りビジネスを拡大していくためには、自社の強みとなる業務をAPI(Application Programming Interface)として外部に公開しながら、他社の強みであるAPIを活用して価値を生み出していかなければならない。そのためには、APIを中心とした迅速かつ高信頼なシステム開発・運用を行うエンジニアリング技術が必要となる。富士通研究所では、これを実現するための技術領域をAPI-Publish & Composeと名付け、複雑化した既存システムから自社の強みとなる機能をAPIとして提供するためのシステム分析技術と、他社から提供されるAPIを活用して素早く安全に新しいサービスをシステム化するための開発・運用技術を研究開発している。

本稿では、API提供者とAPI利用者の双方の立場で必要となる技術に関する富士通研究所の取り組みと、適用事例について述べる。

### Abstract

All kinds of things are becoming digitalized, and the world of digital co-creation that combines these things to create new businesses is expanding. For companies to survive and grow their business in this world, they must create value by presenting their strengths to other companies in the form of application programming interfaces (APIs) and at the same time make use of other companies' strengths by way of APIs. This requires engineering technology for quick and reliable development/operation of API-based systems. Fujitsu Laboratories calls the technical field in which this is achieved, API-Publish & Compose. The Laboratories engage in research and development of system analysis technology for the purpose of making APIs of a company's particular strengths extracted from existing complex systems, as well as development and operation technology for quick and safe systematization of new services using APIs offered by other companies. This paper describes the activities of Fujitsu Laboratories with regard to technologies required by both API providers and API users, and application examples.

---

## ま え が き

インターネットにモノやソフトウェアなどあらゆるものがつながる、新しい世界が始まりつつある。この世界では、これまで人手で行っていた作業も含めて全てがデジタル化され、ソフトウェアによって結ばれ制御されるようになる。そのため、共通言語としてAPI（Application Programming Interface）の重要性が急速に増している。

調査によると、企業におけるAPI利用状況の目安となるAPI管理製品市場の国内年間平均成長率は、2015年～2020年で41.1%に上ると予測している<sup>(1)</sup>。また別の調査では、回答企業の約1/3がAPIの公開によって収益を上げ、市場化までにかかる時間を短縮したと答えている。更に、高度なAPI管理を行っている企業の83%が、競合他社との差別化にAPIの管理が有効であると考えている<sup>(2)</sup>。実際に、米国の先進的な企業だけでなく、日本でも倉庫業者が個人向け保管サービスをAPI化して公開し、これを利用して新たなサービスを提供するサードパーティーとの連携によって、ビジネスを拡大した事例などが現れている<sup>(3)</sup>。

富士通研究所では、ヒト、モノ、ICTを並行・自律・対話的に活用して人に合わせたサービスを創造する、デジタル共創の実現を目指している。これによって、様々な社会・経済・生活の活動のあらゆるシーンでエンドユーザーを支援し、またエンドユーザーが享受可能な価値や利便性を最大化する<sup>(4)</sup>。

このデジタル共創の世界で企業が生き残りビジネスを拡大していくためには、APIを中心としたシステム開発・運用を迅速かつ高品質に行う必要がある。しかし、これを実現するためには、富士通が得意とする従来の業務システム開発とは異なる新たなエンジニアリング技術や手法が必要となる。そこで、富士通研究所ではこのような開発を実現するための技術領域をAPI-Publish & Composeと名付けて、研究開発を推進している。

本稿では、既存システムからAPIを提供するための分析技術や、APIを活用して素早く安全に新しいサービスをシステム化するための開発・運用技術など、APIの提供者と利用者の双方の立場で必要となるシステムエンジニアリング技術について述べる。

## デジタル共創に向けた二つの技術領域

本章では、デジタル共創の世界でビジネスを展開していくための二つの戦略について述べる。

一つ目は、自社の強みを外部に公開してエコシステムを形成する戦略である。ある米国企業が自社の産業機械のメンテナンスノウハウをプラットフォーム化した例<sup>(5)</sup>や、前述の倉庫業者の事例などがそれに当たる。この戦略では、外部公開に向けたビジネスモデルの変革に加えて、自社の既存システムの改革が必要となる。

例えば、これまで自社内からの注文入力だけであった購買システムにおいて、他社がインターネット経由で注文できるようにするためには、他社がインターネットから受注機能呼び出せるように既存システムを変更する必要がある。富士通研究所では、システム変更のためのコストを極力抑えて機能をAPI化するなど、既存システムをデジタル共創の世界で活用できるように移行させるための研究開発に取り組んでおり、これをAPI-Publishと呼んでいる。

二つ目は、他社の強みを取り入れて新たな価値を創出する戦略である。例えば金融業界では、ベンチャー企業が大手金融機関の金融系APIを活用して、資産管理や投資支援などのサービスを提供するFintech市場が既に立ち上がっている。日本では、改正銀行法によってオープンAPI導入が努力義務化されることで、ますますAPIの活用によるビジネス創造の流れが加速する。

しかし、他社が提供するAPIを活用してシステムを構築する際には、新たな課題が発生する。自社ではコントロールできない他社が提供するAPIの品質を管理し変化に追従しながら、安定したシステム運用を実現していかなければならない。富士通研究所では、APIを組み合わせ素早く安全にシステムを構築するための研究開発に取り組んでおり<sup>(4)</sup>、これをAPI-Composeと呼んでいる。API-Publish & Composeの全体像を図-1に示す。

以降の章では、API-PublishとAPI-Composeについて詳述する。

## APIを提供するシステム開発技術：API-Publish

自社の強みである業務をデジタル化するために

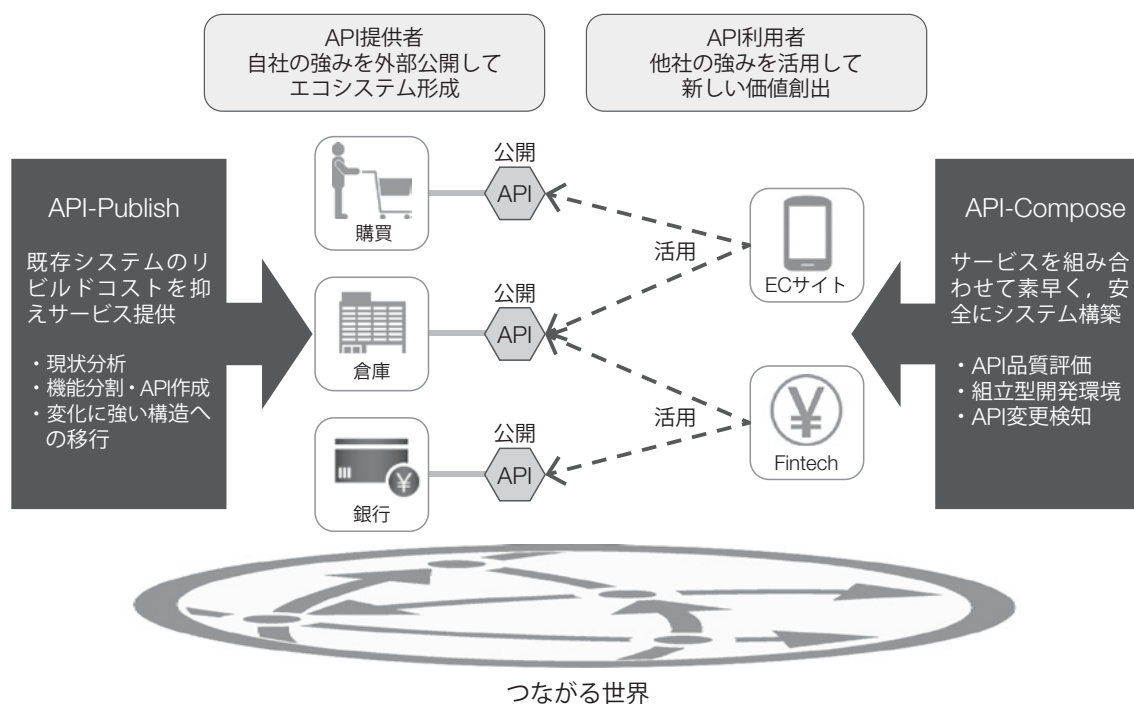


図-1 API-Publish & Composeの全体像

は、受発注などのビジネスそのものをAPI化して、ソフトウェアで制御できるようにすることが必要である。そのためには、既存の業務システムの機能をAPI経由で呼び出せるようにすることが求められる。しかし、APIの提供などによって始まる新しいビジネスは、事前に負荷を見積もることが難しい。そのため、リスクは可能な限り小さく、既存業務に影響が及ばないように立ち上げ、変化するニーズに合わせて徐々に改善していくことが望ましい。その一方で、既存システムは長年の保守によって複雑化しており、変更が困難であったり、変更には多くの工数が必要となったりするケースが多い。そのため、既存システム全体に影響が及ばないように、部分的・段階的に変更する方法が求められている。

API-Publishでは、図-2に示す以下の3ステップによって、既存システムを変化に強いシステム構造へ移行させることを提案している。

#### ● ステップ1：現状分析

まず、既存システムの現状を把握し、API化や保守性向上などの目的に応じて、必要な工数やリスクを見積もる必要がある。<sup>(6)</sup>しかし、長く保守を続けている既存システムは、設計当初に比べて機能

が複雑化する傾向にある。したがって、変更の影響が広範囲に及ぶため、変更規模の見積もりが難しくなる場合が多い。

このような状況に対して、富士通では既存システムのソースコードを分析し、現状を評価する資産分析サービスを提供している。<sup>(7)</sup>このサービスでは、システムや各プログラムの規模、呼び出し構造、および計測した各種メトリクスなどから品質を評価する。また、呼び出し関係を追跡する稼働資産分析によって、使われなくなっているプログラムを洗い出すこともできる。

更に、システムのソフトウェア構造を地図の形で可視化するソフトウェア地図技術<sup>(8)、(9)</sup>による分析・評価も提供している。この地図によって、システム内の各機能間の関係や複雑さなどを把握し、API化したい機能や変更したい箇所とほかの部分の関係を調べながら、変更の影響範囲を見積もることができる。

#### ● ステップ2：機能分割・API作成

現状分析の結果を受けて、次のステップで必要なのは保守性の向上やAPIの追加など、目的に応じた既存システムの変更である。しかし、既存システムの内部は様々な機能とデータが直接参照し

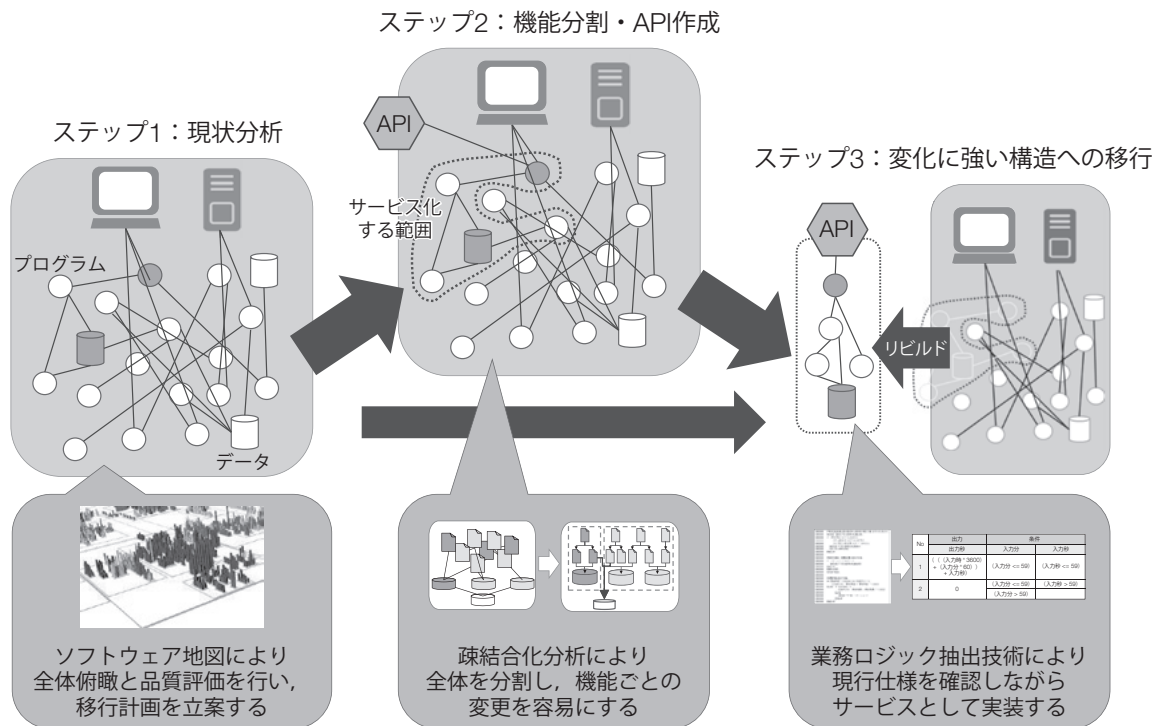


図-2 API-Publishの各ステップ

合っており、どこかに手を入れようとする全体に影響が及ぶ構造になっていることが多い。このため小規模な変更であっても、影響範囲の分析や動作確認のためのテストに多大な工数が必要となり、リリースまでに長い時間を要することになる。

富士通研究所は、このような既存システムの内部を変更しやすくする技術として、疎結合化分析技術を開発した<sup>(10)</sup>。この技術はプログラムとデータの関係性を分析し、関係が強いプログラムとデータをクラスタリングして、クラスタ間にまたがるプログラムの呼び出しやデータの読み書きなどの依存関係が最も少なくなる境界を見つけるものである。

この分析結果を踏まえ、更に業務の観点も加えながら、機能間の依存関係を整理してAPI化することで変更の際にはほかの機能に与える影響を少なくし、外部に機能を提供するAPIが作成できる。

### ● ステップ3：変化に強い構造への移行

既存システムの機能をAPI化して外部に提供すると、既存システムの負荷などに影響が出る可能性がある。この場合、外部に提供したい機能だけを既存システムとは別にサービスとして実装すれば、既存システムを変更せずにAPI化できる。また、既

存システムには影響を与えずにサービスだけを入れ替えることも可能なため、APIの変更が必要になった場合にも迅速に対応できる。

このアプローチを実現するためには、既存システムの仕様と同等な機能をサービスとして新規に実装し、既存システムと並行して稼働させる必要がある。このため、富士通研究所ではプログラム中に実装されている業務ロジックを人間が理解しやすい形で抽出し、再利用を可能にする業務ロジック抽出技術を開発した<sup>(9), (11), (12)</sup>。

これによって、現状分析や機能分割の結果から注目している機能を含む範囲を特定し、その範囲で実装されている業務ロジックを抽出・再利用することで、既存システムと同じ機能を持つサービスの設計が容易にできる。また、こうして抽出した仕様と、利用可能な外部のSaaS（Software as a Service）や社内の共通サービスの仕様との間で、類似や相違を分析することで、既存のサービスとの置き換えも容易になる。

更に、このアプローチによって既存システムの機能を段階的に新システムやSaaSに移行していくことで、最終的にはシステム全体をクラウドネイティブな形態に移行させることも可能となる。

## ● 適用事例

疎結合化分析によって、購買業務のシステムの構造を分割した例を示す(図-3)。2,151本のプログラムから構成されたシステムにおいて、設計当初の命名規約では71の領域に分割されていた。本技術で分析した結果199に分割され、より細かい粒度で分割できることが分かる。分割数が多いにも関わらず、領域間の依存関係の総数は設計当初の84.4%に減少し、更に領域内のデータは外部からの書き込みがないため、依存関係を整理しやすい。

この技術によって得られた領域に沿ってサービス化する範囲を決めれば、外部との独立性を高めることができ、必要となるAPIを作成する負担を軽減できる。

### APIを活用したシステム開発技術：API-Compose

他社が提供するAPIを活用して自社のシステムを構築する場合、システム構築後もAPIは絶えず変化していくという性質から、新たな課題が発生する。図-4は、システムのライフサイクルと、各工程に

おいて新たに生じる課題を整理したものである。例えば、要件定義・設計の工程では、「要件を満たすAPIはあるか?」「信頼できるビジネスパートナーか?」といった「APIをさがす」作業に関わる課題が発生する。

富士通研究所では、図-4に示した課題のうち、システム開発運用技術として重要な「APIを見る」「APIをつなぐ」「システムを直す」の技術開発を進めている。これらの技術によって、他社APIを組み合わせたシステム構築をより早く、より高い信頼性で実現できる。以下に、この三つの技術について説明する。

## ● 「APIを見る」API品質評価

他社が提供するAPIを活用する場合、そのAPIの品質がどの程度信頼できるかを判断することが難しい。APIの品質が分からなければ、ユーザーが考慮すべき機能・非機能要件を認識する必要があるだけでなく、利用方法習得までにかかる期間の見積もり、APIの仕様変更への対応工数見積もりなど、開発運用作業自体の設計も困難となる。

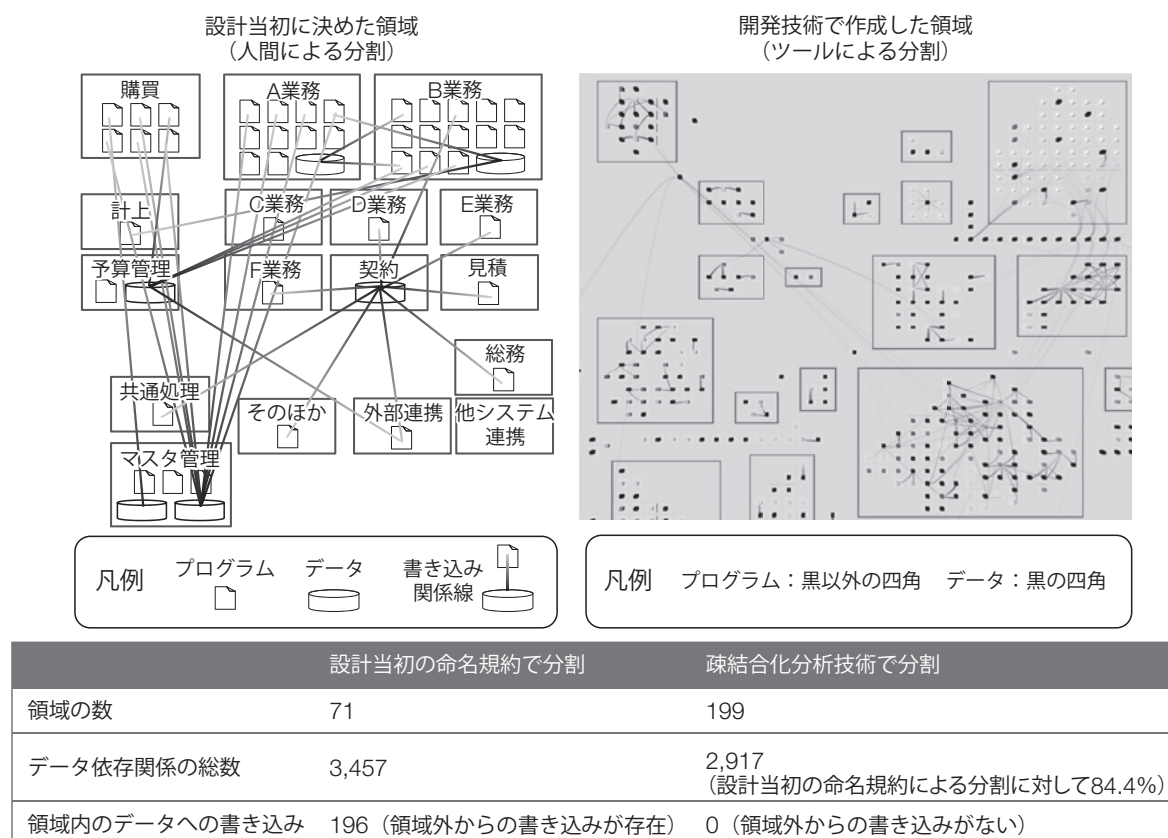


図-3 疎結合化分析の事例

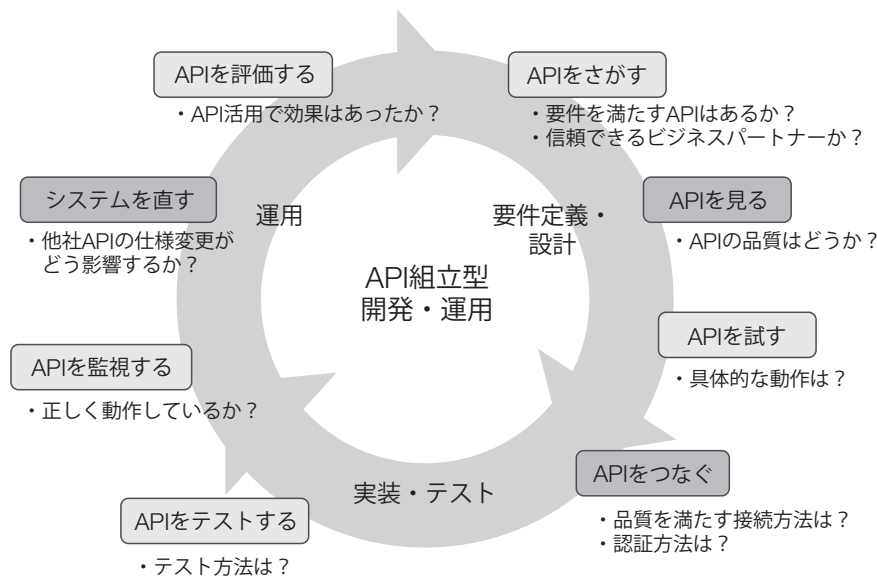


図-4 システムのライフサイクルとAPI組立型開発・運用における課題

これに対して富士通研究所はレスポンス時間や稼働率、平均復旧時間、利用ユーザー数といった既存の評価メトリクスに加えて、習得容易性と仕様安定性を定量評価するメトリクスを開発することで、APIの品質評価を行うことを目指している。

まず、APIの習得容易性を表すメトリクスとして、サンプル網羅値を考案した。これは、APIの仕様を理解するには利用サンプルが網羅的に提供されていることが重要であると考えたためである。

サンプル網羅値は、APIドキュメント内に記載されているAPIのリクエストおよびレスポンスの各パラメーターの仕様と、そのサンプル（“city”パラメーターなら“川崎”など）が対応付けられている程度を数値化したもので、以下の二つの網羅率の調和平均を取ったものである。仕様の理解を助けるサンプルが少ないとこの値が低くなり、習得までに時間がかかると推測することができる。

#### (1) サンプル網羅率

仕様が記述されたAPIとそのパラメーターに対して、そのサンプルが存在する割合。

#### (2) 仕様網羅率

サンプル内のAPIとそのパラメーターに対して、その仕様記述が存在する割合。

また、仕様の安定性を表すメトリクスとして、互換性維持値を考案した。互換性維持値は、APIがバージョンアップされるたびに仕様が後方互換性

を維持しているかを数値化したものである。これは、二つのバージョンのAPIドキュメント間を比較して、後方互換性を維持しているAPIの割合を算出し、それを世代ごとに重み付けして平均化したものである。この値が低いと、APIの変更に対する対処が必要になる可能性が高いことが分かる。

API品質評価では、独自に開発した上記の二つのメトリクスと、四つの既存メトリクスをAPIポータル上でレーダーチャートとして提示することで、個々のAPIの品質を可視化している。これにより、開発者はAPIを活用するときに考慮すべき品質に関する観点を織り込んだ設計が可能となる。

### ● 「APIをつなぐ」API組立型開発環境

デジタル共創では、ニーズに応じてビジネスを迅速に変化させるために、業務フローや業務ルールを素早くシステム化できる開発環境が必要となる。

この目的のために、多くのフローベースの開発環境が提案されている。フロー形式の記述を行うことで、APIを組み合わせることで簡単に自社の業務フローを組み立てることができる。しかし、システムとして安定して動作させるためには、例外処理を実装するなど、業務フロー・ルール以外の作り込みが必要となる。また、リトライ処理やタイムアウト、サーキットブレーカー機能などの信頼性の向上に必要な機能を実装する必要がある。しか

し、これらの機能をフロー記述によって実装することは複雑であるため、多くの工数がかかる。

この問題を解決するために、富士通研究所はこれらの信頼性向上の機能をパターン化し、開発者が記述した業務フローに対して自動補完する技術を開発した。

具体的には、フローエディタとしてOSS（Open Source Software）の開発環境であるNode-REDを採用した。補完したい機能ごとのルールに従ってJSON形式のNode-REDフローの内部表現を変換することで、信頼性向上の機能を実現する部分フローを元のフローに埋め込む。これにより、開発者は業務フロー・ルールの作成に注力してシステムを開発できる。リトライ処理を補完する例を図-5に示す。開発者がWeb APIの呼び出しを含んだフローを記述した後に自動的に補完することで、リトライ処理がフローに組み込まれる。

### ● 「システムを直す」API変更検知

APIは、利用者が予想しないタイミングで変更されることがある。Twitter APIなどのオープンAPIの仕様変更された際には、それまで利用できていたアプリが動作不能となるケースが発生している。しかし、デジタル共創でビジネスを行うケースにおいては、APIの提供者とユーザーとの間でAPIの変更に関するルールを事前に策定しておくことが望ましい。

そこで、富士通研究所はAPIのバージョンアップ

の前に、変更後のAPI仕様をOpen API Specificationの形式で公開することを提案する。これによって、ユーザーはAPIの変更を機械的・自動的に検知して、事前に対応できる。API提供者側においても、Open API Specification形式のAPI仕様を作成することは、そこからソースコードを自動生成できるなど、開発効率化のメリットがあると考えている。

更に、富士通研究所は公開されたAPIの変更によりユーザー側のアプリを変更する必要があるか否かを判断する技術を開発した。具体的には、Open API Specification形式で記述された二つの異なるバージョンのAPI仕様を比較し、対応するエンドポイント、パラメーター、レスポンスの有無などを分析することによって、後方互換性が維持されているかを判断できる。この判断結果を当該APIを活用しているアプリの開発者に通知することで、他社APIの変更に対するスムーズな対応が可能となる。

### ● 適用事例

上述した開発技術の有効性を検証するために、三つの技術をAPIゲートウェイおよびAPIポータルに組み込み、一つのプラットフォームとしてシステム化した。現在、富士通研究所内でそのシステムを試行している。更に、社内開発しているAPI群を題材に、習得容易性や仕様安定性のメトリクス、変更検知技術の有効性評価を進めている。

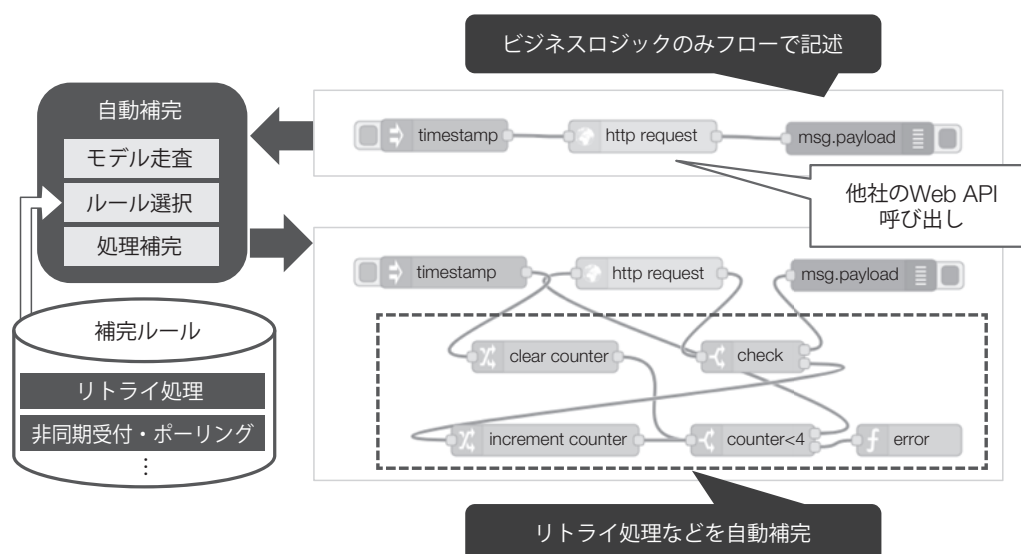


図-5 API組立型アプリ開発環境

## エンドユーザーコンピューティングの実現

デジタル共創が更に進展し、APIであらゆるものが制御できる世界になると、ビジネスとソフトウェアはより一層一体化したものとなる。すなわち、業務部門や経営者が自らAPIを利用したソフトウェア開発を行っていくことが想定される。

本稿で述べたAPI組立型開発環境はその世界につながる取り組みであるが、まだ始まったばかりである。この開発環境には、ICTスキルが十分でなくても簡単にソフトウェアを開発でき、万が一不具合があってもシステム全体を安定して運用できる仕組みが必要となる。

富士通研究所はAPI-Publish & Composeのコンセプトのもと、業務部門や経営者といった業務システムのエンドユーザーによるソフトウェア開発を実現しながら、企業システム全体の信頼性を確保する新たなエンドユーザーコンピューティングを実現していく。

## む す び

本稿では、デジタル共創でビジネスを成長させるために必要となるAPIベースのシステム開発・運用技術であるAPI-Publish & Composeについて、コンセプトと研究開発技術について述べた。

既存システムをデジタル共創につなげる技術は、長年業務システム開発を手掛けてきた富士通ならではのものであり、自社の強みをAPIとして提供するための強力な武器となる。また、APIを活用したシステムを簡単かつ富士通ならではの高い信頼性で実現することも期待されている。

今後は、この期待に応えるため、API-Publish & Compose技術を更に成熟させていくとともに、APIを活用して誰もがソフトウェアを安全に開発できる世界に向けて研究開発に取り組んでいく。

## 参考文献

- (1) ITR：ITR Market View：システム連携/統合モデルウェア市場2016。  
<https://www.itr.co.jp/report/marketview/M16001300.html>
- (2) CA Technologies：CA Technologiesの調査で、日本は世界平均に比べて、APIの活用が出遅れているこ

とが明らかに。

- <https://www.ca.com/jp/company/newsroom/press-releases/2017/ca-technologies-utilization-of-api-is-lagging-behind.html>
- (3) 寺田倉庫：クラウドストレージサービス「minikura」のAPIが特許取得。  
[https://www.terrada.co.jp/wp/wp-content/uploads/2016/05/20160512\\_minikuraAPI-Patent\\_Press-Release\\_JP.pdf](https://www.terrada.co.jp/wp/wp-content/uploads/2016/05/20160512_minikuraAPI-Patent_Press-Release_JP.pdf)
- (4) 河場基行ほか：デジタルイノベーションを支えるソフトウェア技術の将来動向。FUJITSU, Vol.69, No.1, p.93-99 (2018).  
<http://www.fujitsu.com/jp/documents/about/resources/publications/magazine/backnumber/vol69-1/paper16.pdf>
- (5) General Electric：Industrial Data Fabric.  
<https://www.ge.com/digital/predix/industrial-data-fabric>
- (6) IPA/SEC：システム再構築を成功に導くユーザガイド 第2版。  
<https://www.ipa.go.jp/files/000057294.pdf>
- (7) 富士通：アプリケーション運用・保守 FUJITSU Application Management and Outsourcing 資産分析サービス。  
<http://www.fujitsu.com/jp/services/application-services/application-management-outsourcing/apm/services/assetanalyse/index.html>
- (8) 富士通研究所：世界初！アプリケーション資産活用のためのソフトウェア地図作成技術を開発。  
<http://pr.fujitsu.com/jp/news/2012/02/29.html>
- (9) 前田芳晴ほか：業務システムのモダナイゼーション支援技術。FUJITSU, Vol.68, No.5, p.78-84 (2017).  
<http://www.fujitsu.com/jp/documents/about/resources/publications/magazine/backnumber/vol68-5/paper10.pdf>
- (10) 富士通研究所：複雑化した業務アプリケーションを機能ごとに分割・可視化する技術を開発。  
<http://pr.fujitsu.com/jp/news/2017/05/18.html>
- (11) 富士通研究所：業界初！プログラムに埋め込まれた業務仕様を自動抽出する技術を開発。  
<http://pr.fujitsu.com/jp/news/2016/10/11.html>
- (12) 前田芳晴ほか：業務システムのモダナイゼーションに向けた大規模なプログラムからの業務仕様の抽出。

FOSE2016, 2016.

### 著者紹介

---



**上原 忠弘**（うえはら ただひろ）

（株）富士通研究所  
ソフトウェア研究所  
Web API・クラウドネイティブアプリ  
ケーションの開発技術に関する研究開  
発に従事。



**松尾 昭彦**（まつお あきひこ）

（株）富士通研究所  
ソフトウェア研究所  
業務システムの保守や再構築に関する  
研究開発に従事。