

# 「ソフトウェアエンジニアリングの海外動向調査」レポート

2024年10月29日

独立行政法人情報処理推進機構

デジタル基盤センター デジタルエンジニアリング部

ソフトウェアエンジニアリンググループ



## ◆ 背景

Software Is Eating the World といわれるようにソフトウェアが国や企業の競争力の源泉となり、すでに IT 産業だけの課題でなく、産業界全体の競争力に影響を与える深刻な問題になりつつある。そのため、IPAでは、これまで行ってきたソフトウェア開発関連の調査を抜本的に見直し、欧米の最新動向と照らし合わせながら、最先端のソフトウェア関連政策を検討している。

2024年1月にIPAが公開した「2023年度ソフトウェア開発に関するアンケート調査」調査結果では、我が国のソフトウェア開発は人月や SLOC という古い考え方や方法論から抜け出せない状況や、欧米がモデルベース開発手法に取り組む中で依然としてドキュメントベースの開発が根差している状況などが明らかになった。

## ◆ 目的

本調査は、これまでの調査結果を踏まえて、我が国と欧米のソフトウェアエンジニアリングの実態や動向の違いをできるだけ正確に把握するために実施した。

調査結果は、海外のソフトウェアエンジニアリングの最新状況を伝えるとともに、我が国のソフトウェアエンジニアリング改革の方向性を検討する基礎資料として活用する。

- ◆ 調査対象 : 主に欧米の先進的ソフトウェア企業、研究機関、団体
- ◆ 調査委託先 : スケールアウト株式会社 (<https://scaleout.tv>)
- ◆ 調査期間 : 2024年6月～ 9月 (委託先による調査)  
9月～10月 (IPAによる結果整理と追加調査)
- ◆ 調査方法 : 文献・Web調査、オンラインインタビュー (15人)
  - [CMU-SEI](#)の関係者 (複数)
  - [OMG](#)の関係者 (複数)
  - [IFPUG](#)の関係者
  - [ISBSG](#)の関係者
  - [NESMA](#)の関係者
  - [COSMIC](#)の関係者
  - [IEEE](#)の関係者
  - 先進的ソフトウェア企業の関係者 (複数)
- ◆ 仕様書 : [入札公告「ソフトウェアエンジニアリングの海外動向調査」に係る一般競争入札](#)

# ソフトウェアエンジニアリングに関する日本の現状と欧米の仮説

- ◆ 「2023年度ソフトウェア開発に関するアンケート調査」の分析結果から明らかになった日本の現状を踏まえ、日本特有の状況について欧米の仮説を立てた。

日本の現状①： 見積もりは未だ経験と勘に頼っている

欧米の仮説①

ソフトウェアビジネスを戦略的に進めるため、過去のプロジェクトデータに基づく見積もり手法の検討や、新たな開発手法の検討が進んでいる

日本の現状②： エンジニアの地位向上の下、人月ビジネスからの脱却を訴える声が少くない

欧米の仮説②

日本とはビジネスモデルが異なり、欧米では価値ベースビジネスが進んでおり、ここには何らかのソフトウェアの価値評価方法（ノウハウ）があると考えられる

日本の現状③： モデルベースの要求定義は好まれず、ドキュメントベースの要件定義が好まれる

欧米の仮説③

モデルベース開発はsafety-criticalな産業分野への適用から製造分野に浸透し、更にはエンタープライズ分野への関心が高まっている

日本の現状④： コンポーネントベースの「使う」開発よりも業務に合わせて「作る」開発傾向がある

欧米の仮説④

Fit to Standardが定着している欧米では、業務は標準に合わせた上で、作るべきコンポーネントを戦略的に決める文化が醸成されている

- ◆ 日本の現状と欧米の仮説から下記の調査テーマを選出。

## 1. ソフトウェア開発の計画方法

- (1) 見積もり手法
- (2) 見積もりに利用可能な定量管理データ収集
- (3) 開発工期の決め方
- (4) 開発プロジェクト計画の中身や様式

## 2. ソフトウェアの価値評価方法

- (1) クラウドサービスやサブスクリプションのプライシング
- (2) アウトソースに価値ベースで対価が支払われる事例

## 3. モデルベース開発

- (1) 適用の浸透状況
- (2) モデリング言語
- (3) モデル間のインターオペラビリティ

## 4. ビルディングブロック（マイクロサービス、ソフトウェアコンポーネント等）

- (1) モノリスからマイクロサービスへのマイグレーション事例
- (2) レガシーシステムのAPI化事例



## 欧米の仮説

### 1. ソフトウェア開発の計画方法 (見積もり、工期等)

ソフトウェアビジネスを戦略的に進めるため、過去のプロジェクトデータに基づく見積もり手法の検討や、新たな開発手法の検討が進んでいる

### 2. ソフトウェアの価値評価方法

日本とはビジネスモデルが異なり、欧米では価値ベースビジネスが進んでおり、そこには何らかのソフトウェアの価値評価方法（ノウハウ）があると考えられる

### 3. モデルベース開発

モデルベース開発はsafety-criticalな産業分野への適用から製造分野に浸透し、更にはエンタープライズ分野への関心が高まっている

### 4. ビルディングブロック (マイクロサービス、ソフトウェアコンポーネント等)

Fit to Standardが定着している欧米では、業務は標準に合わせた上で、作るべきコンポーネントを戦略的に決める文化が醸成されている

## 調査結果

- 見積もり手法の技術的側面に欧米と日本の違いは無いが、戦略的側面は欧米が進んでいる。
- 技術的側面では、過去のプロジェクト管理データから類推する見積もり手法など、勘と経験に頼らない実績に基づく見積もりの検討や取り組みがある。
- アジャイル開発やコンポーネントベースの開発の見積もり手法は、期待されているものの、まだ目立った事例は出ていない模様。
- 価値ベースビジネスの一例としてSaaSビジネスを調査したところ、BtoBのSaaSビジネスのプライシングでは、これまで利用数ベースが主流であったが、顧客企業の価値に着目した成果ベースのプライシングモデルを適用する企業がブログ等で取り上げられ、注目されている。
- アウトソースにおいて価値ベースで対価が支払われる事例は探せなかった。
- モデルベース開発は、自動車産業や航空機産業のようなsafety-criticalな分野では定着し、今は製造領域に向かっているようであるが、エンタープライズ分野では未だ浸透していない模様。
- エンタープライズ分野の検討はOMGのUAF(Unified Architecture Framework) が注目されており、組織やビジネスレベルのインターオペラビリティにフォーカスしている状況が伺える。
- Agility最優先でモノリスを採用したのちに、マーケット拡張が見通せたところで、マイクロサービス化に移行している事例が多く見つかった。
- 新規開発システム及びレガシーシステムの両方で、顧客起点のサービスがさらに変化することを視野に入れて、戦略的・段階的に、拡張性・適応性への対応を進めている。

# 1. ソフトウェア開発の計画方法

- (1)見積もり手法
- (2)見積もりに利用可能な定量管理データ収集
- (3)開発工期の決め方
- (4)開発プロジェクト計画の中身や様式

## 〔調査結果サマリー〕

- 見積もり手法の技術的側面に欧米と日本の違いは無いが、戦略的側面は欧米が進んでいる。
- 技術的側面では、過去のプロジェクト管理データから類推する見積もり手法など、勘と経験に頼らない実績に基づく見積もりの検討や取り組みがある。
- アジャイル開発やコンポーネントベースの開発の見積もり手法は、期待されているものの、まだ目立った事例は出ていない模様。

# テーマ1 ソフトウェア開発の計画方法

## (1)見積もり手法

### ■ 過去のプロジェクトデータからの類推による新たな手法の検討や取り組みがある

#### (1) 新たな見積もり手法の検討：Capability-Based Software Cost Estimation (CaBSCE)

- CMU-SEIでは、DoDに対して、新たな見積もり手法 “Capability-Based Software Cost Estimation (CaBSCE)”の試行を開始。
- 保有するソフトウェア開発データリポジトリを分析し、類似の機能と特徴を持つソフトウェアを抽出し、コストとスケジュールを見積もる。

参照先：[CMU-SEI, “Capability-Based Software Cost Estimation \(CaBSCE\)”, September 4, 2024](#)

[Anandi Hira, “Capability-Based Software Cost Estimation \(CaBSCE\):Proposing a New Method to Estimate Software Costs”, April 30, 2024](#)

#### (2) 過去のソフトウェア開発プロジェクトデータを活用した見積もりコンサルテーションSaaS事例

- スペインのコンサルティング企業LedaMCでは、保有するソフトウェア開発プロジェクトデータによる見積もりコンサルテーションビジネスを展開している。

参照先：[LedaMC, “SaaS APP for Software Project Estimation”](#)

### ■ 見積もり手法の技術的側面に欧米と日本の違いは無いが、戦略的側面は明らかに違う

- 米国のICEAA International Cost Estimating and Analysis Associationでは、ソフトウェアコスト見積もり認定 (SCEC) 制度を開始している。
- 米国の国防や事業経営の戦略はソフトウェアが中心となるため、見積もり手法は経営側面で体系化され見積もりの専門家育成が始まっている。

参照先：[International Cost Estimating and Analysis Association, “The Certification Program”](#)

### ■ アジャイル開発メトリクスの標準化事例は見つからず

- アジャイルメトリクスは人気が高まっているが、まだ利用ガイドが標準化されていない。アジャイル開発者は、市場比較よりも自己改善に重点を置いている。  
〔インタビュー先〕匿名（見積もり推進団体や研究機関の関係者）

### ■ コンポーネントベースの見積もり手法の検討はあまり進んでいない

- コンポーネントベースのソフトウェアコスト見積もりの事例や手法の研究に関する最近の記事は見当たらない。



# テーマ1 ソフトウェア開発の計画方法

## (2)見積もりに利用可能な定量管理データ収集

### ■ 唯一ISBSGが、メトリクスデータ収集と公開をグローバルに行っている

- 豪州の非営利団体ISBSG (International Software Benchmarking Standards Group) は1997年に設立し、見積もりコンサル事業者等から提供されたデータを公開している。
- 件数：約10,000 (※参考：IPA独自の収集蓄積データは約5000件)
- ウォーターフォール開発プロジェクトのデータが多く集まっているが、アジャイル開発のデータは十分に集まっていない。

参照先：[ISBSG, "A global & independent source of software metrics data for IT projects"](#)

※海外で収集したデータを公開しているのは、ISBSGのみ。

※CMU-SEIは、政府系組織のメトリクスデータを収集しているようだが非公開。

※欧米の見積もりコンサル事業者の中には、独自にデータ収集しているところもあるが非公開。

# テーマ1 ソフトウェア開発の計画方法

## (3) 開発工期の決め方

### ■ 工数積み上げではなく、期間内に何が提供できるかに変わりつつある

- 顧客価値を俊敏に提供するための開発は、計画の考え方が根本的に変わる。工期設定は、工数積み上げベースではなく期間内に何が提供できるかに基づく。
- QCD目標達成を成功と考えるプロジェクトベースのソフトウェア開発の計画方法が社内標準として規定されていた時代ではない。価値重視のソフトウェア開発に移っている。
- 開発工期の決め方で、社内に標準のルールは無い。10年前とは違い、各プロジェクトが柔軟に決めている。  
〔インタビュー先〕 大手グローバルテクノロジー企業の関係者、SaaSプラットフォームを開発するシニアソフトウェアエンジニア

# テーマ1 ソフトウェア開発の計画方法

## (4)開発プロジェクト計画の中身や様式

### ■ ソフトウェア開発の目的がシステム実現から顧客価値に移行すると要求の捉え方が変わる

システム実現を目的とした開発ではウォーターフォール開発の適用が主流であったが、顧客価値追及を目的とするアジャイル開発においては、要求の捉え方が変わり、BRDやMLPというドキュメントが作られている。

#### (1)BRD (Business Requirement Document)

- システム開発の技術的視点で要求を定義するドキュメントではなく、ユーザーエクスペリエンスを重視してビジネス要求を記述するBRDが紹介されている。

参照先 : [asana, "Business requirements document template: 7 key components, with examples", January 21st, 2024](#)  
[Indeed Editorial Team, "Business Requirements Document: Definition and Tips", August 18, 2024](#)

#### (2)MVP (Minimum Viable Product)とMLP (Minimum Lovable Product)

- 仮説検証型開発では、最小限必要な機能 (MVP) を実装してリリースするが、Amazonでは、更にお客様に愛される最小限の機能 (MLP) を定義して実装している。

参照先 : [Doron Katz, "Minimal Lovable Product \(MLP\) #amazon #experience #customerObsession", Jun 17, 2021](#)  
[Komal Singh, "After MVP", Nov 7, 2022](#)

## 2. ソフトウェアの価値評価方法

- (1)クラウドサービスやサブスクリプションのプライシング
- (2)価値ベースで対価が支払われる事例

### 〔調査結果サマリー〕

- 価値ベースビジネスの一例としてSaaSビジネスを調査したところ、BtoBのSaaSビジネスのプライシングでは、これまで利用数ベースが主流であったが、顧客企業の価値に着目した成果ベースのプライシングモデルを適用する企業がブログ等に取り上げられ、注目されている。
- アウトソースにおいて価値ベースで対価が支払われる事例は探せなかった。

# テーマ2 ソフトウェアの価値評価方法

## (1)クラウドサービスやサブスクリプションのプライシング

### ■ BtoB向けのSaaSプライシングは、成果ベースモデルが注目されている

- 企業がROI（投資対効果）をより重視する傾向が強まると、BtoB向けのSaaSビジネスは、顧客企業の成果に寄与することが求められる。

参照先：[Dominic Perrett, Sam Shinner, Jade Kahn, Todd Anstett, "The Rise of Outcome-Based Pricing in SaaS: Aligning Value With Cost", September 13, 2024](#)〔事例1,2〕  
[James Wood, "The spectrum between outcome-based and usage-based pricing metrics: Insights from Intercom's Fin AI and more", April 24, 2024](#)〔事例2〕  
[Bright Reach, "Salesforce Agentforce and the Shift to Outcome-Based Pricing", September 29, 2024](#)〔事例2,3,4〕

#### 〔事例1〕 [Riskified](#)社

- イスラエル、電子商取引の決済時に不正取引を検出するサービスプロバイダー。
- 公正な取引成立時のみ課金され、不正が検出され取引不成立時には課金されない。
- 類似企業にリトアニアの[iDenfy](#)社がある。

#### 〔事例2〕 [Intercom](#) 社

- 米国、BtoCを仲介するメッセージングプラットフォームを提供。
- サービスの1つにAIチャットボットがあり、問い合わせが解決した場合にのみ課金される。

#### 〔事例3〕 [EvenUp](#) 社

- 米国、人身傷害弁護士向けにAIが法的文書を自動作成するサービスを提供。
- 作成した文書件数、文書当たりのページ数で課金される。

#### 〔事例4〕 [Salesforce](#)社のAgentforce

- 米国、Salesforce社が提供するAIプラットフォームで、営業、カスタマーサポート、マーケティングなどのビジネスプロセスの設計を自動化できる。
- 従来の利用者数ベースの課金モデルではなく、AIエージェントへの問い合わせ1回ごとに課金されるモデルに移行している。



# テーマ2 ソフトウェアの価値評価方法

## (2)アウトソースに価値ベースで対価が支払われる事例

### ■ 事例は見つからず

※質の高い個人事業主やフリーランスエンジニアが、開発するソフトウェアの価値に見合った高い報酬を得ている事例をインタビュー等で調査したが、限られた調査期間の中では、事例を見つけることが出来なかった。

# 3. モデルベース開発

- (1)適用の浸透状況
- (2)モデリング言語
- (3)モデル間のインターオペラビリティ

## 〔調査結果サマリー〕

- モデルベース開発は、自動車産業や航空機産業のようなsafety-criticalな分野では定着し、今は製造領域に向かっているようであるが、エンタープライズ分野では未だ浸透していない模様。
- エンタープライズ分野の検討はOMGのUAF(Unified Architecture Framework) が注目されており、組織やビジネスレベルのインターオペラビリティにフォーカスしている状況が伺える。

# テーマ3 モデルベース開発

## (1)モデルベース開発の浸透状況

### ■ safety-critical領域では既に浸透、デジタルツインさらには、エコシステムに適用

- 2023年12月にDoDは、[DoDI 5000.97“Digital Engineering”](#)を公開し、MBSEやモデリング言語の使用を基本方針としており、システム開発のコミュニケーション手段を「ドキュメント」から「モデル」に移行することを指示している。
- モデルはシステムを表現するだけでなく、推論、分析、シミュレーションにも不可欠。
- モデルには、原子力発電所や航空機などの既存システムのモデルと同期して、予知保全や異常検出を行うデジタルツインの概念も含む。  
〔インタビュー先〕 CMU-SEI

### ■ 即応性と適応性が求められる分野での浸透状況は不明

- 頻繁に変化する追加要求があり、それに柔軟に応じることを求められる開発では、MBSEのような重いプロセスは使われていない。  
〔インタビュー先〕 大手グローバルテクノロジー企業の関係者、SaaSプラットフォームを開発するシニアソフトウェアエンジニア

### ■ OMGのSysML v2が、ゲームチェンジャーとして期待されている

#### (1)SysML v2 の概要

- 独自開発したメタなモデリング言語（Kernel Modeling Language）がベースになっている。v1は、UMLベースを拡張していた。
- グラフィカル表記に加えて、テキスト構文に基づく形式言語でモデルを表記できる。
- 共通インタフェースAPIが用意される。
- 2024年内に公開予定。

参照先： [Sanford Friedenthal, "SysML v2 Basics", January 28, 2024](#)

#### (2) 現バージョン SysML v1 の利用状況と評価

- 2024年1月にIPAが公開した[「2023年度ソフトウェア開発に関するアンケート調査」調査結果](#)によると、日本におけるシステムレベルのモデリング言語SysML v1の利用状況は、UMLに比べると低い。
- 欧州では、制約のあるSysML v1よりもThales社のCapellaに人気があったようだが、Capella利用者もSysML v2との接続に関心を持っている。

〔インタビュー先〕 OMG

- OMGサイトに、INCOSEがSysML v1の改訂に際しThales社のCapellaとの比較を行った資料が公開されている。

参照先： - [Torrance INCOSE IW Meeting - January 2017 SysML v2 RFP Working Group Meeting](#)

- [Stéphane Bonnet, "Arcadia/Capella: Looking back at our implementation issues CONSIDERATIONS FOR SYSML V2", January 28, 2017](#)

## ■ ビジネスレベルのモデリング手法では、OMGのUAFが注目されている

### (1)UAF(Unified Architecture Framework)の概要

※V1.2@2022年7月が最新。初版V1.0@2017年10月発行

- Part1:ドメイン メタモデル (DMM)、Part2:UAF モデリング言語(UAFML)で構成されている。
- 下記のArchitecture Frameworkを統合している。
  - 米国国防総省アーキテクチャ・フレームワーク (DoDAF)
  - 英国国防省アーキテクチャ・フレームワーク (MODAF)
  - NATOアーキテクチャ・フレームワーク (NAF)
- 軍事領域の要求事項は、ビジネス分野の要求事項の90%をカバーすると考えられている。
- デジタルエコシステム実現のためには、組織構造や事業構造のモデル化が必要になるためUAFが注目されている。

参照先 : [OMG, "Introduction to 'Unified Architecture Framework®\(UAF®\)'"](#)

### (2)もう一つのArchitecture Framework

- Architecture Frameworkは、オープングループのTOGAF version1.0が1995年に公開されている。

参照先 : [The Open Group, "Welcome to the Digital Edition of the TOGAF Standard"](#)

- オープングループは、2019年のArchitecture Forumにて、TOGAFとUAFの比較と、TOGAFのUAFへのマッピングを議論している。

参照先 : [The Open Group, Denver 2019: Proceedings – Architecture Forum: The TOGAF® Standard to OMG UAF Mapping](#)



# テーマ3 モデルベース開発

## (2)モデリング言語の状況 BPMN

### ■ OMGのBPMNは完成し、CMMNやDMNがBPMNを補完している

(1)BPMN Business Process Model and Notation (ビジネスプロセスモデルと表記法)

※V2.0.2@2014年1月以降更新なし。

参照先: [OMG, "Business Process Model & Notation™ \(BPMN™\)"](#)

(2)CMMN Case Management Model and Notation (ケース管理モデルと表記法)

※V1.1@2016年12月以降更新なし。

参照先: [OMG, "Case Management Model and Notation™ \(CMMN™\)"](#)

(3)DMN Decision Model and Notation (意思決定モデルと表記法)

※V1.6@2024年9月が最新。初版V1.0@2015年9月発行後、更新継続。

参照先: [OMG, "Decision Model and Notation™ \(DMN™\)"](#)

(4)SDMN Shared Data Model and Notation (共通データモデルと表記法)

※V1.0 beta 2@2024年2月発行後、正式版発行に向けて更新中。

参照先: [OMG, "About the Shared Data Model and Notation Specification Version 1.0 beta 2"](#)

- OMGのビジネスプロセスモデリング手法では、基本となるBPMNは完成済みであるが、プロセスフローを事前に決定できない場合（医療、消防、警察などの緊急事態）の管理などは、BPMNは適さないため、CMMNがこれを補完できる。
- 同じ領域で BPMN、CMMN、DMN を使用する場合、SDMNで共通データをモデル化する。
- 最近では医療分野の組織モデルに注力している。

[インタビュー先]OMG

### ■ 組織レベル、システムレベル、ソフトウェアレベル各モデル間のインターオペラビリティが課題

- 組織（ビジネス）、システム、ソフトウェアそれぞれのレベルのモデリングは、まだ言語が共通化されていない。
- モデリングの研究機関や団体は、共通のモデリング言語によって、組織（ビジネス）のモデルからシステムのモデル、ソフトウェアのモデルへとシームレスに移行できるように検討している。
- OMGでは、ソフトウェア、システム、組織（ビジネス）のモデリング言語として、それぞれUML、SysML、UAFを公開している。
- SysML v2では、モデルに接続するための汎用 API が用意される。
- また、SysML を OWL などのオントロジー定義言語にリンクして、インターオペラビリティを強化するための取り組みも進行している。

〔インタビュー先〕 CMU-SEI

### ■ MBSEは、システムモデリングに限らず参照されている

- モデリング対象がシステムレベルからソフトウェアや組織を含めた範囲に広がると、MBSEは、システムモデリングに限らず参照されている。
- CMU-SEIでは、MBSE(Model-Based Systems Engineering)に対して、ソフトウェアが強調される場合は、model-based **Software engineering**と呼んでいる。

〔インタビュー先〕 CMU-SEI

# 4. ビルディングブロック

## (マイクロサービス、ソフトウェアコンポーネント等)

- (1) モノリスからマイクロサービスへのマイグレーション事例
- (2) レガシーシステムのAPI化事例

### 〔調査結果サマリー〕

- Agility最優先でモノリスを採用したのちに、マーケット拡張が見通せたところで、マイクロサービス化に移行している事例が多く見つかった。
- 新規開発システム及びレガシーシステムの両方で、顧客起点のサービスがさらに変化することを視野に入れて、戦略的・段階的に、拡張性・適応性への対応を進めている。

# テーマ4 ビルディングブロック

## (1)モノリスからマイクロサービスへのマイグレーション事例

### ■ 新規開発モノリスからマイクロサービスへのマイグレーションを戦略的に実施

下記のECサイトやECモールは、戦略的にいち早くマーケットシェア獲得のために、Agility最優先でモノリスを採用している。マーケット拡張が見通せたところで、スケーラビリティ、機能追加・変更、セキュリティ対応、障害の最小回避等を可能にするために、マイクロサービス化に移行している。

#### 〔事例1〕 [Netflix 社](#)

- 米国、インターネットでストリーミングサービスを提供

参照先 : [Netflix, "Completing the Netflix Cloud Migration", 12 February 2016](#)

#### 〔事例2〕 [Uber 社](#)

- 米国、スマートフォンアプリを通じて世界中でタクシー配車・フードデリバリー・貨物輸送などを展開

参照先 : [Adam Gluck, "Introducing Domain-Oriented Microservice Architecture", 23 July 2020](#)

#### 〔事例3〕 [Etsy 社](#)

- 米国、ハンドメイド作品やヴィンテージ商品のECモール

参照先 : [Stefanie Schirmer, "API First Transformation at Etsy - Concurrency", 6 September 2016](#)

#### 〔事例4〕 [Amazon Prime Video](#)

- 米国、Amazon社が提供するストリーミングサービス

参照先 : [AWS, Prime Video Boosts Scale and Resilience Using Amazon DynamoDB , 2018](#)

# テーマ4 ビルディングブロック

## (2) レガシーシステムのAPI化事例

### ■ レガシーシステムにAPIゲートウェイを提供して他システムとの連携を進めている

APIゲートウェイから着手して、他システムから単純なインターフェースで呼び出せる形にしながら、コンポーネント化と拡張性への対応をじっくりと進めている。

#### 〔事例1〕 [Finologee 社](#)

- ルクセンブルク、デジタル金融プラットフォーム運営会社
- 既存システムへのアクセスをAPIで提供するサービスプラットフォーム
- 欧州が制定する「決済サービス指令」PSD2に対応して、銀行が口座情報取得や決済指示をAPIで提供するためのプラットフォームを構築
- IBM API Connect®ベースで開発

参照先：[IBM, “Fast-track digital compliance for the financial industry”](#)

#### 〔事例2〕 [TINE 社](#)

- ノルウェー、ノルウェー最大の乳製品の生産・販売・輸出会社
- 既存システムを維持しながらAPI Gatewayにより統合システムを開発
- 複雑に絡み合っている複数のレガシーシステムをモダナイズするために、新たに「統合センター」を開発。IBM API ConnectをGatewayとして、順次統合を進めている。既存システムはオーナーの責任で運用し、複雑化したインターフェースを戦略的にAPIで整理している。
- IBM API Connect®ベースで開発

参照先：[IBM, “Untangling a complex value chain”](#)



IPA