

## Web API を活用した EDI 機能の検討

藤井章博<sup>i</sup> 中山真樹<sup>ii</sup> 田中浩一<sup>iii</sup> 長村州浩<sup>iii</sup>

企業情報システムにおける業務アプリケーションのインターフェースとして、Web API (Application Programming Interface) を活用するシステムの利用が広がっている。そこで、Web API が前提とした異企業間の電子商取引、特に EDI (Electronic Data Interchange) の実現方法に関して検討する。本稿では、Web API を利用する EDI の形態を特に「Open EDI」と呼ぶ。このプラットフォーム上で、EDI の実装単位となる情報の参照を「Cloud Object」と呼ぶ構造で規定する。このことによって、特定のプラットフォームに依存しないデータ連携のための汎用の枠組みを規定することを目標としている。

### EDI Support System over Web API platform

Akihiro FUJII<sup>i</sup> Maki NAKAYAMA<sup>ii</sup>  
Koichi TANAKA<sup>iii</sup> Kunihiko NAGAMURA<sup>iii</sup>

Recently Web API (Application Programming Interface) is widely utilized for the service interfaces in enterprise applications. In this article, the possibility of applying Web API to electronic commerce, especially to EDI (Electronic Data Interchange), is discussed. We propose architecture called “Open EDI” as a platform for such activities. And the notation of “Cloud Object” is introduced that is the structured operational unit for EDI. We aim the system works as a general web applications without specific operational environment for the data-exchanges.

### 1. はじめに

流通関連企業による EDI (Electronic Data Interchange) の導入の経緯を振り返ると、規模の大きささまざまな取り組みが存在し、活用の度合いは様々である。そうした中で技術的な先進性・新規性を特長とする事例としては、例えば UN/CEFACT (貿易簡易化と電子ビジネスのための国連センター) と OASIS (構造化情報標準促進協会) が推進する ebXML 規格の策定とその普及への努力が注目される。日本国内において ebXML 普及を先導していた ECOM (次世代電子商取引推進協議会) が平成 19 年 3 月に公開した「ebXML による次世代 EDI 促進報告書」の記述によれば、将来の ebXML 仕様の普及状況に関する可能性の高いシナリオとして、「Web サービスなど、他の次世代 EDI 標準との技術的融合が進み、双方向の相互運用が容易になる。」と述べている。[1]

異企業間での情報共有を困難にする大きな原因は、情報共有システム構築のコストが大きい点である。コストは、大別して EDI 実装のための基盤整備やプラットフォームとアプリ導入コストなど技術的要因に基づくものと、企業同士の製品コードの統合やマッピングなど主にビジネス的要因に拠るものがある。

昨今の情報通信技術の進展に目を向けてみると、Web API (Application Programming Interface) 技術を活用してデータ連携を実現する方法が幅広く利用されるようになっており、特に前者の観点から異企業間のデータ連携を比較的低コストで実現できる状況にあるといえる。[9][10]

本研究では、Web API 技術を活用することによって、商品流通に携わる複数企業が比較的容易に EDI を実現する方式について検討する。具体的には、商品マスターのデータベースに対して Web API によりデータ共有できるプラットフォームを構築する場合を想定し、データに付随する取引情報を関連する複数企業から Web API を介して参照できる形で EDI システムを実現する。本稿では、そのためのアーキテクチャを検討し、異なる企業間における EDI のための情報処理の単位を Cloud Object という設計概念で規定する。このような規定をオープンに定めることによって特定のプラットフォームに依存しない EDI の実現を模索している。

<sup>i</sup> 法政大学理工学部応用情報工学科 <sup>ii</sup> ネクストオブジェクト(株) <sup>iii</sup> (有) マルチパラダイムシステムズ

## 2. 技術動向と関連研究および研究上の課題

### 2.1 技術動向

現在、Google や Amazon など大手の商用サイトは、自社の提供する情報サービス、例えば Google マップのような地図データベースの情報を Web API で利用者に提供している。このように、Web API を組み合わせるソフトウェア開発の技法が広まっている。これは、Yahoo がユーザ向けマッシュアップ用ツールである Pipes を 2007 年に発表したことが契機となり、これによって可能となるデータ連携機能を利用してビジネスアプリケーションを構築する動きが着実に進行している。特に、この技法を企業における情報システムやビジネスアプリケーションに適用させることを「エンタープライズ・マッシュアップ」と呼ぶ場合がある。

こうしたデータ連携の登場によって、ソフトウェア開発およびソフトウェアに基づくサービス提供の形態は今後大きく変化し、利用者の側からも新しいビジネスアプリケーションが多数登場してくる可能性がある。なぜならば、利用者は特に高度なソフトウェア開発の知識がなくとも、Web API として提供されている既存のソフトウェア部品を組み合わせることによって、利用者自身にとって便利な機能を短期間に容易に開発し、公開することができるようになっていくものと考えられる。こうした動向を「エンドユーザ開発」と呼ぶ場合もある。[2]

### 2.2 関連研究

例えば、欧州の FAST プロジェクトは、欧州委員会の ICT 政策からの資金提供を受け、企業情報システムのあり方を研究している。2008 年 3 月にはじまり、2011 年 3 月末まで実施される。その予算規模は、全体では 550 万ユーロ（約 7 億 2 千万円）である。これは 2004 年から 2008 年まで 1520 万ユーロ（約 19 億 8 千万円）の規模で実施された「SecSE (Service Centric Software Engineering)」研究プロジェクトの後継であり、複数企業のサービス連携を目指すプラットフォームを構築するという試みである。サービスを中心にアプリケーションを開発し、利用するための有効な方法やツールなどを開発することを目的としている。

FAST プロジェクトの研究成果の一つに、SAP の研究所がまとめた論文として、[4] があげられる。ここでは、Web API を活用する企業間のデータ連携をいくつかのパターンに分類しモデル化している。一方、IBM で検討しているプライベートクラウドの環境でも Web API によるデータ連携が議論されている。[7]

Web API からの入出力を組み合わせることによって、EDI といった高度なデータ連

携を実現するためには、API との入出力を一連のフローで記述する必要がある。例えば、W3C の規定する XProc では、XML 文書処理する相互運用可能な標準的手法が定義され、その普及の努力が払われている。また、Yahoo がエンドユーザ向けに提供している Pipes などのプラットフォームの利用拡大は、ユーザによるデータ連携機能を利用するビジネスアプリケーションの構築を促しているといえる。[5][6]

### 2.3 研究上の課題

昨今、SaaS(Software as a Service) や「プライベートクラウド」という用語によって Web API を活用することによる企業情報システムの新しい可能性が議論されている。そこで、研究上の課題としては、実務的に有用な具体的な解を事例として提示することが重要であろう。さらに、そうしたシステム設計における汎用的な考え方を示すことも必要であると考えられる。

我々は、機械部品の流通分野における EDI に対して、具体的に上述したような技術動向を踏まえた開発を行ってきた。[9]本稿では、実装の具体についての詳細を述べることはせず、以下では、Web API によって EDI を実現する際のシステム設計に関するひとつの考え方を提唱することを目的とする。

## 3. Open EDI アーキテクチャ

本研究では、例えば機械部品流通分野など具体的な応用分野での EDI の実現を目指している。具体的な分野を想定していても、Web API を利用したデータ連携の方法に対してできるだけ汎用性のある設計思想をとる。

### 3.1 システムを構成する要素

まず、Web API を活用して EDI システムを構築するための基本的な情報処理環境のモデルを規定する。システムは、一般的な Web による情報処理環境を想定している。システムは次の三つの要素から構成されていると仮定する。

①データベースに蓄積された商品情報などのリソース、②Web API を活用するための機能ユニット、③EDI を実現するためのマッシュアップ機能。それぞれの要素は、次の処理を司るとする。このような情報処理環境は、いわゆる「プライベートクラウド」と呼ばれる情報処理環境の前提となるシステム構成要素である。[4][7]

- ① リソース：データベースとその管理を行う。商品データベースの参照情報は API を介して上位のミドルウェア階層とのインターフェースを持つ。例えば製品に関して異なる企業で異なるコード体系で運用している場合、それらのコード共通化するために「コード」を API によって公開する機能もこの階層に位置づける。
- ② 機能ユニット：Web API を活用するためのミドルウェアの機能を提供する。Google の API を利用する場合は、「Pipe」に相当する。例えば文献[4]においては、「Gadget」と呼んでいる。API を組み合わせ、処理を行うことにより、EOS 機能や EDI を実現する。ミドルウェアでの情報の授受は、XML によって記述される。
- ③ マッシュアップ：ビジネスアプリケーションを提供する機能を提供する。個々の Web API 利用のための仕様・規約に従って、それらを組み合わせてアプリケーションを構築する。構築されるアプリケーションの運用には、新たに Web 上で利用者との情報の入出力インターフェースを持つ場合が多いので、こうした Web 上の処理は、この要素の役割とする。

### 3.2 データ連携のシナリオ

一口に「EDI の実装」といっても処理の内容や実装の複雑さの程度には大きな幅がある。複雑なプロセスを議論することはひとまず避け、分かりやすい事例として、例えば、次のようなシナリオにおける関連企業間のデータ連携を考えてみる。

- (1) 流通におけるバリューチェーンの上部に位置する「メーカー」は、自社の製品情報を Web API で参照可能な形で提供する。
- (2) 販売を担当する「ディーラー」は、複数のメーカーの商品を取り扱うため、自社ブランドのカタログを有する。カタログを構成するための情報は、「メーカー」の提供する製品情報に関する Web API を参照する。
- (3) ディーラーは、製品に関する在庫管理、価格管理、配送・物流などの処理を実施する。本稿で報告の対象とする EDI 機能は、付録に挙げた機能の一部である。

このシナリオでの具体例は、メーカーが提供する製品データベースの Web API を提供する。その一方で、ディーラーは、販売管理のための在庫情報、価格情報、決済機能などの機能を提供する Web API を提供しそれらの連携が必要である。

現実の流通プロセスはより複雑な場合もあり、このシナリオだけでは網羅できない。しかし、この例では、Web API を EDI で活用するために必要な基本的な要点を含んでいる。具体的には、①商品の購買者も含めると3つ以上の組織が電子商取引に関与している。②取引に必要な情報は、二つ以上の組織が提供する情報によって成立する。③これらの情報は、上述したリソースが得られる。④これらのリソースの参照は Web API によって行われる。⑤Web API を組み合わせるためのプラットフォームを利用する。

以下では、これらの Web API を利用するためのデータ連携の単位を「Cloud Object」と呼ぶ設計概念によって構造化することを試みる。その構造単位に必要な要件は、①～⑤の状態に含まれているといえる。

### 3.3 「Cloud Object」の定義

本稿で提唱する Web API によるオープンな EDI のしくみを「Open EDI」と呼んでおり、プライベートクラウドのような実装環境を想定しているため、その上で参照されるオブジェクトを Cloud Object と呼んでいる。

図1に Cloud Object の定義の骨子を述べる。Cloud Object は、Resource と Operation を有する。それぞれは、図に述べる機能を具備する。

まず、「Cloud Object」とは古典的な“モノ”を表すオブジェクトではなく、「業務上関連のある“行為”(=ロジック)を束ねたものに加えて、その行為に必要なデータをモデル化したもの」だとみなす。例えば「受注業務」という“行為”を起点としたモデルである。

つぎに、参照系たる"Resource Complex Query"の Operation と、更新系たる"Resource Lifecycle Management"の Operation は次の特徴をもつ。Complex Query の方の API は Resource が直接提供し、Lifecycle Management の方は、Function がそのロジックを装備して提供する。

## 4. 考察

本節では、提案する Open EDI の新規性を関連システムとの比較において論ずる。まず、「分散オブジェクト指向技術」と呼ばれるソフトウェア工学技術として、世界規模の業界団体 OMG (Object Management Group) が規定する CORBA (Common Object

Request Broker Architecture) 仕様が存在する。ここでは、ネットワーク上に分散するソフトウェアの部品を結合するという考え方がとられる。他に Java プログラムを対象とした Java RMI(Remote Method Invocation)などの規格が存在する。これらを利用するシステムでは、実行ランタイムが分散している場合の解決策を与えることが目的であるので、異なる組織に跨るデータ連携は予め主目的としては想定されていない。

次に、昨今急速に利用者が増大している「salesforce.com」などのプラットフォームでは、SaaS(Software as a Service)の考え方を前面に出して、サードパーティが提供するサービスを共通のプラットフォームの上で提供することで便宜を図るというものである。ここで、想定されているサービスの形態は、基本的には個々に完結しているものであって、相互のデータ連携を積極的に意図しているものではない。

一方、「Yahoo Pipes」などの考え方では、あるサービスの出力を別のサービスの入力に利用する。これによって、異なるサービス主体同士の連携が考えられているが、サービスの提供者については特にモデル化されていない。

以上に対して本稿で提唱する Open EDI においては、EDI に関連する企業各社や各部門の API サービスの設計自体を CloudObject という概念でモデル化する点に新規性があり、それら Cloud Object が相互に提供する API サービスの「マッシュアップ」によって、全体でひとつの EDI アプリケーションを構成する設計思想となっている。

## 5. むすび

東大阪市に本拠を置くねじ企業間情報処理研究会では、大阪、京都、奈良にあるねじ商社約 30 社が集まり、共通の EDI システムを構築することを目的に、研究会を設立し、1997 年(平成 9 年)の以来、ねじ統一名称および関連事項のコード付け、プロトコルの制定、通信手段の決定、EDI 導入に伴う基幹システム構想の具体化などを手がけてきた。現在、商品名のレベルで約 3400 種類、25 万アイテムの業界統一コードを制定して活用している。[3]

本稿で提案する Open EDI の仕様に基づいて、具体的にはこのような既存の EDI 環境を対象に実装を検討している。研究開発の目標は、「1. はじめに」で述べたように EDI を実現するための技術的側面におけるコスト削減である。ビジネスに起因する要因であるデータの統一や、コードの統合あるいはマッピングの問題を OpenEDI のアーキテクチャ上でどのように扱っていくかは、分散サービスというアーキテクチャの特性を踏まえつつ検討中である。[8]今後、実装の評価はこのような観点から行っていく予定である。

Cloud Object の定義		
この CloudObject で使用するデータ。URL で表される。	Internal Resource	Resource の内、ローカル (□同じドメイン) に存在。
	External Resource	Resource の内、リモートに存在する。
この CloudObject が提供している API。業務機能。	Resource Direct Access Operation	Operation の内、Resource に対する単純 CRUD 操作を提供する。参照系 (R) と更新系 (C/U/D) を有する。更新系はプロダクション環境下では通常アクセス制御で呼び出しが禁止されているべき。
	Resource Simple Query Operation	Operation の内、単一 Resource に対する単純問い合わせ操作を提供するもの。参照系 Operation。
	Resource Complex Query Operation	Operation の内、Resource に対して、SQL SELECT 並に様々な問い合わせをする。CloudObject = Function で「使用する」と宣言されている範囲の Resource にアクセスできる。この Operation は CloudObject=Function を“配置”するとデフォルトで無条件で装備される。参照系 Operation。
	Resource Lifecycle Management Operation	Resource Lifecycle = 業務プロセスの進行を装備しているもの。例えば「注文を取り消す」、「発送する」など。この Operation はロジックを明示的に記述する必要がある。更新系 Operation。

図 1 Open EDI リソース定義

## 参考文献

- 1) 次世代電子商取引推進協議会、「ebXMLによる次世代 EDI 促進報告書」2007年3月
- 2) 水越 悠太, 服部 哲, 速水 治夫、「ノンプログラミングによる Web アプリケーション開発支援システムの提案」情報処理学会グループウェアとネットワークサービス研究会、GN75-1、2010年3月(神奈川工科大学)
- 3) ねじ企業間情報処理研究会「会社ごとに異なるばらばらの EDI コードをネットワーク EDI で統一管理」、中小企業 IT 経営力対象 2008、<http://www.itouentai.jp/case/index.html>
- 4) Till Janner et. al. “Patterns for Enterprise Mashups in B2B Collaborations to foster Lightweight Composition and End User Development” , IEEE International Conf. on Web Service, 2009
- 5) Alessandro Bozzon et. al “A Conceptual Modeling Approach to Business Service Mashup Development” , IEEE International Conf. on Web Service, 2009
- 6) L. Ardissono, et al. “ From Service Clouds to User-centric Personal Clouds” , IEEE International Conf. on Cloud Computing, 2009
- 7) Hong Cai, et al., “Customer Centric Cloud Service Model and a Case Study on Commerce as a Service” , IEEE International Conf. on Cloud Computing, 2009
- 8) 米国ハーバード大学法学部事例研究資料、「Mashups Interoperability and eInnovation」<http://cyber.law.harvard.edu/interop>
- 9) 藤井章博、「機械部品流通 EAI 向け API プラットフォーム」、情報処理学会電子社会基盤研究報告、EIP44-3、2009年6月
- 10) 藤井章博「広がる Web API の活用 - マッシュアップの幅広い可能性 - 」科学技術動向、文部科学省科学技術政策研究所、No. 106、2010年1月