

Web サービス連携を可能にする Web サービス フローコントローラ

Web Service Flow Controller Enabling Orchestration of Web Services

大森 洋一 望月 克人

■ OMORI Yoichi

■ MOCHIZUKI Katsuhito

東芝テック(株)は、Web サービス連携のための標準言語 BPEL4WS (Business Process Execution Language for Web Services) の実行エンジン及び開発環境から成る Web サービス フローコントローラを開発し、当社製ミドルウェア “CrossMission™” 上に搭載した。

この製品は、BPEL4WS の標準機能による Web サービス連携に加え、従来資産である Windows® (注1) プロセスとも連携可能という特長を持つ。更に、Web サービスのセキュリティ標準である WS (Web Services) -Security のパスワード認証及び暗号化の機能に対応している。これらにより、社内及び社外のシステムとの連携が容易になり、SOA (Service Oriented Architecture) に基づいた連携システムを容易に開発できる。

Toshiba TEC Corp. has developed a Web service flow controller, containing an execution engine and a development environment, based on a standard Web service orchestration language called BPEL4WS (Business Process Execution Language for Web Services). It runs on our CrossMission™ middleware.

The Web service flow controller enables cooperation with legacy Windows® processes. A business process designer can design business logic without knowing the difference between Internet-based Web services and legacy processes. The Web service flow controller also supports both the password certification and encryption functions of the Web service security standard WS-Security. Therefore, our proposed mechanism allows Internet and intranet systems based on service-oriented architecture (SOA) to be easily developed.

1 まえがき

SOA (Service Oriented Architecture) と呼ばれるシステム構築手法が注目されている。これは、サービスと呼ばれるソフトウェア部品を組み合わせることで大規模なシステムを構築する手法である。SOA では、サービスを実装している言語に依存せず、サービスのインタフェース情報を使ってシステムを構築する。このため、システム構築後にサービスの中身や提供者が変更されても、サービスの機能とインタフェースが変わらない限り、システムのその他の部分に手を加える必要はないという利点がある。

SOA を実現するための具体的な技術として、Web サービスが広く知られている。Web サービスは、WSDL (Web Services Description Language) と呼ばれるインタフェースを使い、SOAP (Simple Object Access Protocol) と呼ばれるプロトコルで、XML (eXtensible Markup Language) で記述されたデータをやり取りする仕組みである。

この Web サービスを連携させるための標準言語として、BPEL4WS (Business Process Execution Language for Web

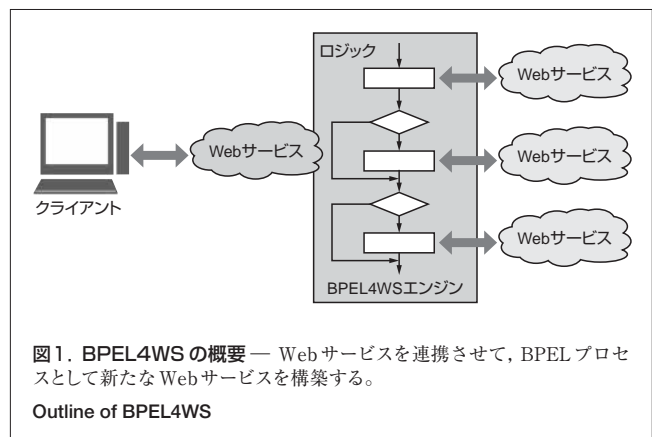


図1. BPEL4WS の概要 — Web サービスを連携させて、BPEL プロセスとして新たな Web サービスを構築する。

Outline of BPEL4WS

Services) が存在する。BPEL4WS は、マイクロソフト社や IBM 社などによって仕様が定められ、標準化団体 OASIS (Organization for the Advancement of Structured Information Standards) によって標準化作業が行われている。BPEL4WS を使うと、図1に示すように複数の Web サービスをワークフローから呼び出す仕組みを構築することができ、その統合されたサービスを一つの Web サービスとして公開することが可能となる。そこではワークフローのロジックや、使用する Web サービスを変更することによって、システムの変更を柔軟に行うことができる。Web サービスベースの

(注1) Windows は、米国 Microsoft Corporation の米国及びその他の国における登録商標。

SOAをより簡単に実現するための標準仕様がBPEL4WSであるとされる。

BPEL4WSの機能を東芝テック(株)製ミドルウェアの“CrossMission™”⁽¹⁾と組み合わせることにより、CrossMission™で可能だったシステム内あるいは企業内のレガシーサービス間の連携に加え、Webサービスを使った企業間の連携も可能になる。これにより、レガシーサービスとWebサービスの両者を同一に扱えるSOAベースのフレームワークとなる。ここでは、当社が開発したWebサービスフローコントローラの技術的な特長について述べ、実際の適用例と将来に向けての展望を紹介する。

2 Webサービスフローコントローラの特長

Webサービスフローコントローラは、BPEL4WSバージョン1.1に準拠するワークフロー実行エンジン及び開発環境であり、.NET(マイクロソフト社が提供するアプリケーション動作環境)対応ビジネスフレームワークであるCrossMission™上で動作する。CrossMission™は、SOAPを利用したメッセージングフレームワークにより、イントラネット内のアプリケーションロジック(Windows®プロセス)を実行制御し、効率的な疎結合システムを実現できる。一方、Webサービスフローコントローラは、BPEL4WSの仕様に基き、Webサービス呼出しや、メッセージ送受信、データ操作、補償処理などのワークフロー操作が記述されたBPELスクリプトを解釈し実行することにより、インターネット上のWebサービスを連携させる。イントラネットを対象としたCrossMission™と、インターネットを対象としたWebサービスフローコントローラを協調させることにより、SOAに基づいたシステム内及び企業内連携だけでなく、企業外のサービス利用や企業間連携も可能となる。また、BPEL4WSの標準仕様には盛り込まれていないが、当社独自の拡張仕様として、WS-Securityに準拠したWSE(Web Services Enhancements)のセキュリティ機能に対応し、パスワード認証及び対称鍵暗号方式AES(Advanced Encryption Standard)による暗号化通信をサポートしている。

ワークフロー開発環境として、ツリービューによるグラフィカルなBPELスクリプトエディタとともに、BPELスクリプトをWebサービスとして公開するためのWebサービス配置ツールを提供している。BPELスクリプトはXMLにより表現されるため、プラットフォームに依存しない。また、Webサービス配置ツールではBPELスクリプトとWSDLを指定することで動的にWebサービスを作成し、WebサーバIIS(Internet Information Services)上に配置することができる。これにより、システム構築者は、アプリケーション開発に関する知識を必要とせず、ワークフロー設計に集中してシステム構築ができる。

3 Webサービスフローコントローラの構成

Webサービスフローコントローラは、開発環境と実行エンジンから成る。

3.1 開発環境

BPELスクリプトを記述し、Webサービスとして公開するための開発環境として、以下のツールを提供している。

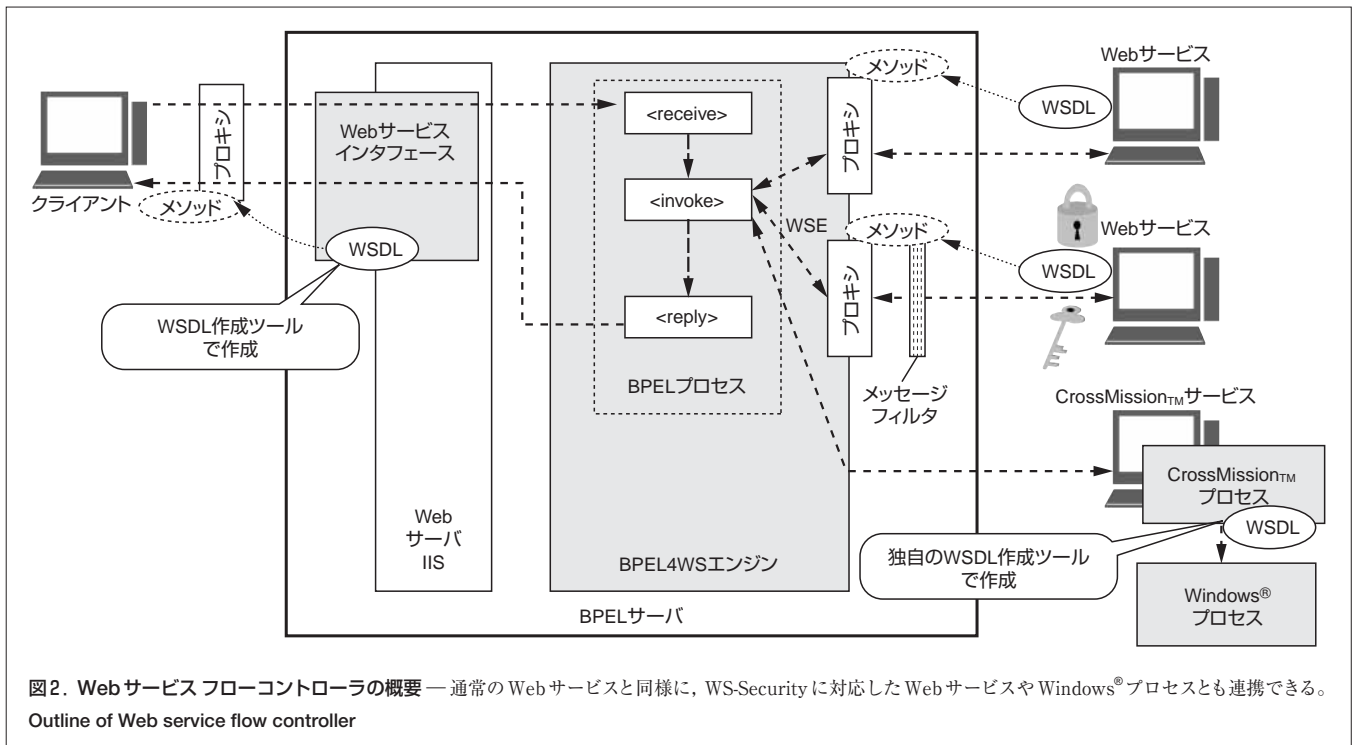
3.1.1 BPELスクリプトエディタ BPEL4WSはメッセージ受信・送信やWebサービス呼出し、データ操作などの処理をアクティビティと呼び、そのアクティビティの組合せにより複雑なビジネスロジックを定義する。BPELスクリプトエディタでは、ツリービュー上に、各アクティビティに対応したアイコンを配置、ドラッグ&ドロップ操作によりワークフローを記述する。また、BPELスクリプトエディタは入力支援機能として、記述中のアクティビティに連動してアクティビティの説明や記述エラーに関する情報を表示するダイナミックヘルプ機能、BPELスクリプトの文法検証機能などを備えている。

3.1.2 WSDL作成ツール WSDL作成ツールは、BPEL4WSで記述して公開するビジネスロジックにアクセスするための、インタフェース情報を設定するツールである。ウィザード形式のツールに従って各種項目を入力することにより、XMLで表現されるWSDLを作成することができる。

また、CrossMission™プロセスを一般のWebサービスと同一に扱えるようにするため、独自のWSDL作成ツールを提供している。これにより、システム構築者はBPELスクリプト記述時にサービスの違いを意識する必要がない。

3.1.3 Webサービス配置ツール 作成したBPELスクリプトをBPELプロセスとして公開するためのツールである。ここで、BPELプロセスとは、ビジネスロジックを実行させる実体である。Webサービス配置ツールでは、Visual Basic®^(注2)、Visual C#®^(注3)、Visual J#®^(注4)の各プログラム言語に対応したWebサービスのテンプレートを用意している。選択されたテンプレート、BPELスクリプト、WSDLを基にVisual Studio®^(注5).NETのIDE(Integrated Development Environment)を利用してWebサービスを作成し、IIS上に配置する。一連の動作によりBPELプロセスのインタフェースがWebサービスに動的に追加されるため、システム構築者はソースコードを編集する必要はない。また、WSEのWS-Security機能を利用して認証と暗号化通信を行うためのテンプレートも用意しており、簡単にセキュリティ機能を追加できる。WebサービスはVisual Studio®のプロジェクトの形で提供されるため、配置後にソースコードを修正するこ

(注2)、(注3)、(注4)、(注5) Visual Basic, Visual C#, Visual J#, Visual Studioは、米国Microsoft Corporationの米国及びその他の国における登録商標。



とで、セキュリティ機能を強化することも可能である。

3.1.4 Web サービス フローモニタ BPELプロセスが正常に動作しているかを監視するためのツールである。Webサービスフローモニタは、Webサービスフローコントローラから送信されるログをウィンドウ上に表示し、BPELプロセスに異常を発見した場合は、対象のプロセスを選択して停止することができる。

3.2 実行エンジン

BPELプロセスは、プロセス開始のためのメッセージを受信すると、BPELスクリプトとして記述されたロジックに従って、フロー処理を開始する。以下に、BPEL4WSを構成する代表的なアクティビティを挙げ、その処理内容を説明する(図2)。

3.2.1 メッセージ受信アクティビティ “receive”

receiveアクティビティは、公開されたWebサービスインタフェースを介して、メッセージを受信するアクティビティである。Webサーバがクライアントからメッセージを受信すると、実行エンジンは対応するreceiveアクティビティへメッセージを渡す。

3.2.2 メッセージ送信アクティビティ “reply”

replyアクティビティは、receiveアクティビティでクライアントから受けたメッセージに対する応答処理を行うアクティビティである。replyアクティビティは、応答メッセージをWebサービスインタフェースを経由してクライアントに返す。

3.2.3 サービス呼出しアクティビティ “invoke”

連携先のWebサービスやCrossMission™プロセスを呼び出すための処理である。Webサービスを呼び出す場合、

対象のWSDLからWebサービスにアクセスするためのプロキシを動的に生成、このプロキシを通じてWebサービスを呼び出す。また、パスワード認証と暗号化通信機能、CrossMission™プロセス呼出し機能は、本来のBPEL4WSにはない仕様のため、invokeのパラメータに独自仕様のオプションを追加することにより実行可能である。セキュリティ機能をオプションとして選択した場合、Webサービス実行時にWSEの機能によりメッセージフィルタを経由ようになる。メッセージフィルタでは、メッセージ送信時にはパスワード情報の付加及びメッセージ暗号化を、メッセージ受信時にはパスワード認証とメッセージ復号を実施する。CrossMission™プロセス呼出しをオプションとして選択した場合は、プロキシを生成せず、WSDLの情報を基に対象のCrossMission™プロセスを直接実行する。

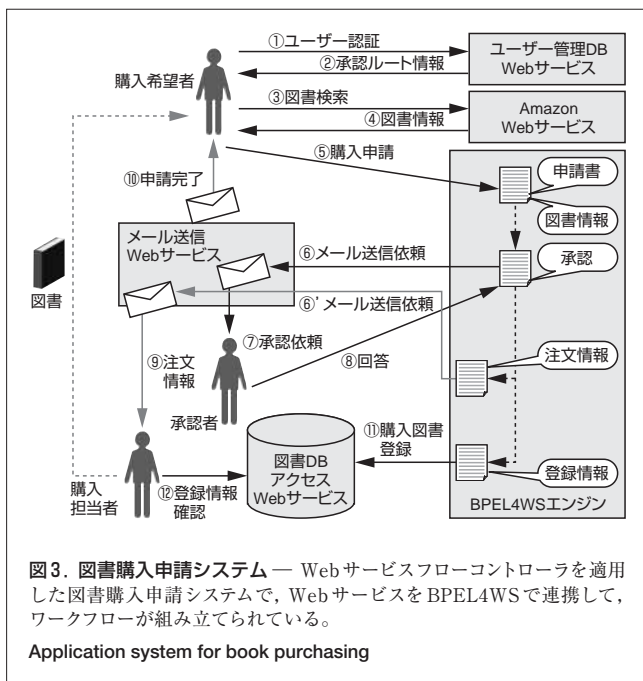
4 Web サービスフローコントローラの適用例

Webサービスフローコントローラを社内で利用している図書購入申請システムに適用し、運用評価を実施している。従来のシステムでは、購入したい書籍の情報を申請者が記入する必要があった。通常は、出版社や書店のWebサイトで図書情報を検索し、それを申請書に転記していた。その申請書を受理した購入担当者は、購入業者向けの電子ファイルに転記して購入業者に送り、図書データベースにも転記する必要があった。このように、人手による転記作業が発生し、効率的でないやり方になっていた。そこで、Amazon社の

Webサービスで図書情報を取得し、その情報をベースにして転記作業が必要ないようにした。

具体的な動作を以下に示す(図3)。

- (1) まず、購入希望者によって、ユーザー管理DB(データベース) Web サービスで認証が行われる(①)。登録されているユーザーの場合、認証成功とともに承認ルート情報を取得する(②)。
- (2) 次に、購入希望者によって、Amazon社が公開しているWeb サービスを利用して、購入希望の図書の検索が行われ(③)、対象の図書情報を反映させた図書購入申請書を作成する(④)。
- (3) 購入申請書に購入希望理由などの情報が入力され、図書購入申請が開始されると(⑤)、メール送信Web サービスにメール送信依頼が行われ(⑥)、最初の承認者宛てに承認依頼メールを送信する(⑦)。
- (4) 承認者によって回答されると(⑧)、メール送信Web サービスにメール送信依頼を行い(⑥)、次の承認者へ承認依頼メールを送信する(⑦)。このワークフローでは、購入担当者を含めて承認者の数を変更でき、この承認手続きを人数分繰り返す。
- (5) 承認者の承認がすべて通ると、注文情報を作成し、メール送信Web サービスにメール送信依頼を行い(⑥')、購入担当者に注文情報メールを送信する(⑨)とともに、申請者へ申請完了メールを送信する(⑩)。
- (6) 注文情報のメール送信依頼完了後、図書DBへの登録情報を作成し、図書DBアクセスWebサービスへ購入図書登録情報を送り(⑪)、図書DBに情報を登録して、



ワークフローを完了する。購入担当者は必要に応じて登録情報を確認することができる(⑫)。

- (7) 承認者の誰かによって否認された場合、メール送信Web サービスにメール送信依頼を行い(⑥)、購入希望者に否認された旨の通知を送って(⑩)、ワークフローを完了する。

このシステムでは、申請を開始する前にAmazon Web サービスを利用して、購入希望者が申請書を作成する際に、記入がめんどうな図書情報、すなわち図書名、価格、著者名、出版社名やISBN(国際標準図書番号)といった情報を入力する手間を省くことができる。

BPEL4WSで構築されていることによって、例えば承認者の数を変更するといった、ワークフローを変更することが簡単にできる。また、図書購入申請システムで使われているユーザー管理DB Web サービスやメール通知Web サービス、図書DB アクセス Web サービスなどのモジュールを利用して、会議室予約などのワークフローを簡単に構築できる。

5 あとがき

ここでは、BPEL4WSに対応したWebサービスフローコントローラを開発し、効率的に図書購入申請システムを構築できたことを示した。今後は、システムを実際に運用して得られた知見を元に、更なる改良を図っていく予定である。

また、社内製品へ適用しやすいように、必要な機能をアダプタとして追加していくことを検討している。具体的には、流通システムへの適用を目指して、ebXML (electronic business XML) や、EDIINT AS2 (Electronic Data Interchange-Internet Integration Applicability Statement 2) といったインターネット EDI のプロトコルの実装を計画中である。

OASISで標準化作業中の仕様“WS-BPEL 2.0”にも対応して、エンジンの仕様を強化しながら、より上位のレベルのモデリングから実装に導けるような仕組みも検討している。

文献

- (1) 平野和順, ほか. XML Web サービス対応ビジネスフレームワーク“Cross-Mission™”. 東芝レビュー. 58, 2, 2003, p.15-18.



大森 洋一 OMORI Yoichi

東芝テック(株) 技術本部 プラットフォーム開発センターグループ開発第二担当主務。Web サービス関連ミドルウェアの設計・開発に従事。
Toshiba TEC Corp.



望月 克人 MOCHIZUKI Katsuhito

東芝テック(株) 技術本部 プラットフォーム開発センターグループ開発第二担当。Web サービス関連ミドルウェアの設計・開発に従事。
Toshiba TEC Corp.