

SOAとWeb 2.0の展望

日本アイ・ピー・エム株式会社
ソフトウェア開発研究所
STSM(Senior Technical Staff Member)
吉田 洋一 Yoichi Yoshida

2006年7月25日、日本アイ・ピー・エム株式会社(以下、日本IBM)箱崎事業所で行われた、WebSphere®のCTO(Chief Technical Officer:最高技術責任者) Gennaro Cuomoの講演「SOAとWeb 2.0の展望」についてご紹介いたします。彼は、2006年5月にIBMコーポレーション(以下、IBM) Fellowに就任しました。

SOAの次への課題

変化の激しい環境下の企業活動を支えるITソリューションを構築する考え方として提唱される、SOA(Service Oriented Architecture: サービス指向アーキテクチャー)、実際に多くの企業でSOAに基づいたソリューションが構築されつつあり、組織をまたがった業務の効率アップや企業活動の変化への対応力の向上に貢献しています。その結果、企業の中には、SOAの考え方に基いて作られた数多くの情報資源やアプリケーション資源がサービスという形で整備されつつあります。

次のステップとして、企業の中の知識労働者が担当する非定型業務の効率化が挙げられます。このような非定型業務では、ビジネスのニーズに基づいて企業内外の多様な情報を加工・組み合わせることで、企業の意思決定に関連したアウトプットを出すことを目的としています。その内容は、環境の変化に応じて常に変わるため、従来の手法や技術ではITソリューションとして開発することが難しかった分野でもあります。

このため、スプレッドシートやワープロなどのオフィスアプリケーションが多く使われてきましたが、それらはインターネットやSOAの世界とは親和性が弱いという現実があります。Web 2.0は、SOAに基づいてつくられた多彩なサービス資源やインターネット上の資源を業務に生かすことができる技術であると考えられます。SOAを用いてつくられた企業のITシステムの中で、Web 2.0が知識労働者の業務をどのように変えていくかについてお話ししたいと思います。

Web 2.0とは

まず、Web 2.0の流れについて簡単に述べます。

本来、Webの技術はURI(Uniform Resource Identifier)というWeb上のアドレスによって指し示されるリソース(データ)に対して、そのデータの取得と表示、送信、削除を行うことが基本になっています。Webサーバーに対しては、画面単位でRequestの送信とRespondの受信を行います。近年、Webにおける情報が爆発的に増え、情報提供の方法の多様化が進むに連れて、多数の技術が生まれてきました。大きな流れをまとめると、以下のようになります。

- Web上のリソースに関して、メタな情報とデータやコンテンツを分離して配信する。
- Web上のリソースに対しても、一般のデータベースのアプリケーションのように、CRUD(Create, Read, Update, Delete)でアクセスできるようになってきた。
- Webブラウザで表示されるコンテンツのデータを非同期にサーバーから取得し、動的にコンテンツを変更することが可能になってきた。

また、これらの技術を容易に組み合わせることのできる技術や手法が提唱され始め、新しいタイプのアプリケーションが生まれつつあります。

このような流れの中で、Web 2.0を支える代表的な技術について簡単に紹介したいと思います。

RSS(Rich Site Summary, RDF Site Summary) / Atom

RSSIは、Webサイトの更新情報を簡単にまとめて配信するための文書フォーマットの総称です。RSS自身にも幾つかのバージョンが存在し、バージョン間の競合も一部発生しているため、AtomがRSS 2.0の後継として提唱され始めました。現在はIETF(Internet Engineering Task Force)に標準化の作業が引き継がれています。

REST(Representational State Transfer)

RESTは、当初は以下に示すような原則に基づいたアーキテクチャーの原則と制約の集まりでした。

- URIを用いてリソースを一意に識別する、ステートレスなクライアント/サーバー間のプロトコル。
- GET, PUT, POST, DELETEを用いた、リソースに対する操作。
- HTMLやXMLを用いた、情報(ファイル)の送受信。

このような原則に合わせたWebサービスシステムは、しばしばRESTfulといわれます。WSDL(Web Service Description Language)で定義されたWebサービスシステムに比べて、インターフェースの定義に関する厳密さはありませんが、それがかえって一般ユーザーにとって使いやすいものになっています。

Ajax(Asynchronous JavaScript + XML)

Ajaxは、JavaScriptを使ったサーバーに対するXML文書の非同期通信とHTMLページの動的変更を用いた、Webアプリケーションの構築技法の総称です。従来のHTMLページ単位で作られたWebアプリケーションに対し、Ajaxを使うことでHTMLページの遷移とは非同期にXML文書をサーバーから受信し、HTMLページを動的に変更することで、より使いやすいユーザーインターフェースを提供できるようになります。

Mashup

Mashupは、ある特定の技術を指すものではなく、Web 2.0の世界におけるアプリケーションもしくはサービスの作り方に関して、その方向性を示します。すなわち、RSSやAtomを通して配信されるWebからの情報や、RESTなどを使用して得られる情報を、JavaScriptやPHPなどのスクリプト言語を用いて新しいサービスを作ることを指しています。

Web 2.0とSOA

それでは、SOAに基づいて構築されたITソリューションの下で、前述したWeb 2.0が知識労働者の生産性をいかに向上させるかについて、具体的に説明いたします。

損害保険会社の社員が台風の進路予想に関するニュースを見て、被害に伴う損害保険の支払いについて予測を立てることにしました。

- ・ 台風の進路上の洪水警報地域の郵便番号から、その

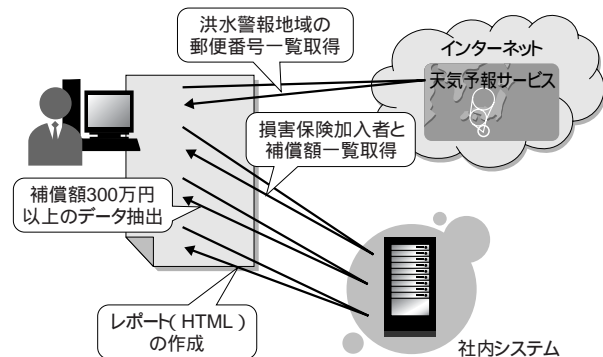


図. 損害保険の支払い予測シナリオ

地域の損害保険の加入者と補償額の一覧を取得。

- ・ 一覧から補償額が300万円を超える契約を抽出し、支払い予測に関するレポートを作成して所属長に提出。
- ・ 予報が変わるたびに定期的に再計算を行ってレポートを更新し、再提出。

このシナリオを実装するアプリケーションを作るためには、以下のようなWeb 2.0に基づいたサービスが必要となります。

- (A) 洪水警報の出ている地域の郵便番号の一覧を返すWeb上のRESTサービス。
- (B) 郵便番号をキーにして保険加入者と補償額の一覧を返す、社内のSOAシステムが提供するRESTサービス。
- (C) (B)のRESTサービスで返されたデータを一時的に保管する社内のRESTサービス。
- (D) 上記のサービスを組み合わせ、保管されたデータから条件に合うものを抽出し、それに対する操作を記述できるようなスクリプト言語。

この社員は、まず(D)で述べられているスクリプト言語を使い、(A)で述べられているRESTサービスを呼んで、洪水警報の出ている地域の郵便番号の一覧を取得します。その一覧に対して(B)のRESTサービスを呼び、加入者と補償額の一覧を取得し、(C)のRESTサービスを使って一時保管をします。その一覧から補償額が300万円以上のものを抽出し、補償額の支払い予測の結果をHTMLページとしてWebサーバーに保管し、そのURLをEメールで上司に通知します。

以上のようなアプリケーションを作成し、予報が変わるたびに実行します。もしくは、Ajaxを使用して非同期に郵便番号の一覧を取得し、残りの処理を実行するように修正することも可能です。このように、SOA化された社内システムによって提供されるサービスとWeb上にあるさまざまなRESTサービスとを、スクリプト言語で組み合わせる(Mashup)用いることで、繰り返し使えるアプリケーションをつくるのが容易になり、業務の生産性を飛躍的に上げることができるのです。

Web 2.0は、一般のWeb上での新しいWebアプリケーション構築のための技術として見られることがよくありますが、いまお話ししたように、SOA化されたITシステムを持つ企業にとっても有用な技術群ということが出来ます。今後はIBMとしても、Web 2.0の技術がいかに業務の効率化を進め、大きな効果が得られるかについてお客様と一緒に考えていき、新しいソリューションを幅広く提供していきます。