

マッシュアップと Web サービス API

～ Web サービスの組み合わせによる新たな価値の創造～

牧野 浩之, 天白 進也

Hiroyuki MAKINO, Shinya TEMPAKU

1 はじめに

最近, Web2.0 という言葉を頻繁に耳にするようになった。Web2.0 とは, 従来のようにコンテンツを見せるだけという仕組みからかけ離れて, 既存技術を駆使して表現手法を工夫したり, ユーザ参加型の精神を取り入れたりすることをいう。現時点では明確な定義づけはなされていないが, Web2.0 にはいくつかの概念があり, ネット全体が Web2.0 のトレンドへ徐々に進化してきている。マッシュアップは Web2.0 を表現するための手法の一つである。マッシュアップを積極的に使うことで, 多くの興味深いサービスが登場しネット上を賑わせている。

本報告では, マッシュアップとはどのようなものであるか, また, それを実現するための Web サービス API に関して紹介するとともに, 今後の展望について述べる。

2 マッシュアップ

マッシュアップとは, そもそも音楽業界の用語であり, アーティストや DJ が複数の曲を組み合わせて, 一つの曲に合成することを意味していた。転じて, Web の世界では, 複数のソース (情報源) を組み合わせて, 一つのサービスを作ることを意味している。Fig. 1 のように一つのサイトの内部システム上では複数のサービスが動いている。

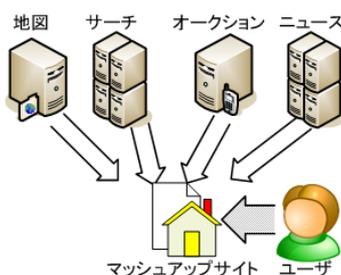


Fig. 1 マッシュアップ (出典: 自作)

2.1 マッシュアップが流行した背景

マッシュアップが流行した背景には, Google や Amazon などが相次いで Web サービス API (Application Program Interface) を公開したことが挙げられる。Web サービス API は, 人間のためのユーザインタフェースではなく, ソフトウェアのためのインタフェースであり, Web

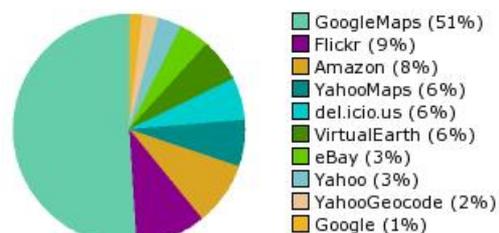
サービス API を通して取得した情報をプログラムで処理をすることが容易になった。様々なサービスで Web サービス API が公開されており, 地図や検索だけでなく, アルバム, カレンダー, 宅配便, オークション, ニュース, 映像配信, ショッピングなどほぼ全てのサービスを網羅しているといっても過言ではない。ユーザは情報の受信者にとどまらず, 取得したデータを加工して利用することにより, 容易に膨大な情報の発信者となることが可能になったのである。

2.2 マッシュアップのメリット

前節で述べたように, ユーザは Web サービス API を用いることで, 短時間, 低コストで充実したコンテンツを持ったサイトを構築することが可能であり, また, 様々な API を組み合わせてマッシュアップすることで, サービス提供者が思いつかないような面白いサービスが創造できる可能性がある。このことは, 当然, 情報を利用する側にとってメリットがあるが, 同時にそのデータの利用価値も上がるので, 情報の提供者側, 利用者側双方にメリットがあるといえる。

2.3 マッシュアップを使ったサービスの例

Fig. 2 はマッシュアップサイトで使われている API の割合を示したものである。これによると, Google Maps API が実に 50%以上を占めている。地図データは非常に有用性が高いことが分かる。



ProgrammableWeb.com 04/15/06

Fig. 2 マッシュアップグラフ (出典: 参考文献 [1])

たとえば, HousingMaps (<http://www.housingmaps.com/>) は Google Maps の API と不動産情報の API を組み合わせて, 地図上で不動産物件の検索が行える。Google Maps の API が提供される以前は不動産会社が地図を用意してその地図上に物件をプロットすることなどコ

ストが大きすぎたが、マッシュアップによりそのコストが払拭されたといえる。

3 Web サービス

Web サービス API は SOAP や REST といったプロトコルで提供されることにより、開発における負担が圧倒的に軽減された。Amazon は Web サービスに SOAP と REST インタフェースの両方を採用しているが、開発の容易さから利用者の 85% は REST インタフェースを採用している。

3.1 SOAP

SOAP とは Simple Object Access Protocol の略でありデータのやり取りには実際には単なるテキストデータである XML 文書を用いることで、お互いの環境に依存しないデータ交換を実現している。また、エンベロープと呼ばれる付帯情報が付いた XML メッセージを、通信プロトコルで交換している。通信プロトコルには HTTP だけでなく、SMTP や FTP も利用可能になっている。SOAP はアクティビティ指向になっており、Fig. 3 に示すようにアクティビティに 1 つずつオペレーションがあり、オペレーションが中心的存在となるものである。



Fig. 3 SOAP(アクティビティ指向)(出典: 自作)

3.2 REST

REST とは REpresentational State Transfer の略であり、WWW の実装にあたってリソースを特定する URI と、リソースにアクセス、操作するためのプロトコルとして、シンプルな HTTP (POST, GET, PUT, DELETE) を使用するという仕組みである。REST はリソース指向であるため、Fig. 4 に示すように、共通のオペレーション (POST, GET, PUT, DELETE) の一群はリソースへのアクセスを支援するものである。

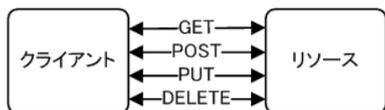


Fig. 4 REST(リソース指向)(出典: 自作)

3.3 SOAP と REST の利用

SOAP は機能面では REST に勝るが、REST に比べて構造が複雑であるため、敷居が高い。従って、SOAP はビジネスロジックなど、システムの頑健性が求められる

場合、REST は簡単に面白いサービスを作りたい場合などユーザのニーズによって使い分けられている。

3.4 Web サービスの現状

現在提供されている Web サービス API は様々な仕様になっており、開発者はそれぞれの仕様を理解して開発しなければならないとなり、負担が大きい。

また、Web サービス API 提供元のサーバがダウンすると、それを利用しているサービス全体に影響するといったデメリットがある。

4 今後の展望

技術面では、API の提供が当然となり、複数の API が複数の資源を共有しながら一つの高性能なシステムとして動くようになり、それに伴って規格の統一化が進むと考えられる。SOAP と REST はそれぞれのメリットを活かして共存していくが、REST は開発の敷居が低いため今後も高い普及率を維持していくと考えられる。

さらに SOAP や REST はシステムの疎結合を促進すると考えられる。たとえば、ユーザインタフェースの変更を行おうとするとビジネスロジックの修正も余儀なくされてしまうといった状況が発生するのはアプリケーションを構成する要素が密結合となっているためであるが、SOAP や REST によって構成要素を散在させることで、保守性も向上する。

サービス面では、Web サービス API の機能拡張が顕著化し、文字や画像だけでなく動画配信といったユーザの情報発信がさらに活発になると考える。現状の REST では POST と GET がほとんどであるが、PUT や DELETE での機能拡張も行われるようになると思う。そうすると、1 つのデータベースを複数のサービスが共有できるソーシャルデータベースのようなものが可能となる。

また、マッシュアップをさらに容易に実現させるために、ネット上に存在する Web サービスの検索、照会を行うポータルサイトのようなものが必要になってくると考える。ディベロッパーのニーズを的確にマッチングできるかどうか今後の注目すべきところである。

参考文献

- 1) ProgrammableWeb Mashup Dashboard
<http://www.programmableweb.com/mashups>
- 2) リソース指向 Web サービスとアクティビティ指向 Web サービスを比較する
http://www-06.ibm.com/jp/developerworks/webservices/041119/j_ws-restvsoap.html