

マッシュアップツールの開発とそれを用いた東京都防災マップの構築

山本剛士[†] 小山智久[†] 杉田達朗[†] 塚本享治[‡]東京工科大学[†] バイオ・情報メディア研究科[‡]

1. はじめに

地震、洪水などの際の避難施設の情報は、今日の都市化した東京都において重要なものである。避難施設をあらかじめ確認しておけば、迅速な避難を可能とし、二次災害発生予想箇所を避けることが可能である。

避難施設情報は主に自治体単位で情報が整備・開示されており、それらの情報を地図化したものを防災マップと呼ぶ。

しかし、東京都においてはそれらを総合した防災マップは存在しないため、地図の内容や表現形式がそれぞれにおいて異なり、公共的な情報が画一でないという問題がある。そこで本研究では、各自治体がウェブ上で公開する避難施設情報を総合的に収集し、Google Maps を用いてマップ化するシステムを開発した。

2. 避難施設情報収集における問題点

本研究において、避難施設情報の収集には各自治体がインターネット上で公開する Web サイトから行うことにした。しかし、Web サイトからの情報収集には大きな問題が存在する。

それは、収集できるデータが HTML であることである。HTML はそもそも、ブラウザでレンダリングされることを前提とした言語であり、提供される情報そのもののセマンティクスを示していない。その結果、画一的な情報の収集が困難である。また、HTML は、その特性上、また歴史上、非常に曖昧な記述でもブラウザは独自の解釈を行ってしまうために、HTML の仕様にすら沿っていない Web ページも数多く存在し、データの抽出を困難にさせている。

そこで、本研究では、HTML から情報を抽出し、抽出したデータを連結し、新たなサービスに対してリクエストを行うといったサービスの連結と変換を行うための言語を開発することでこの問題に対処した。

3. マッシュアップ言語の開発

本研究では、上記問題に対処するために、サービスとサービスの連結を行う(俗にマッシュアップと言われる)専用の言語を作成するというアプローチで解決を図った。

3.1 スクリーン・スクレイピング

本研究における情報収集媒体はインターネット上のサービス、つまり HTML であることから、HTML から必要な情報のみ抽出する必要がある(Screen Scraping)。しかし HTML は曖昧であるばかりか、仕様に沿っていない、間違った HTML であってもそれが、許可されてしまう歴史的な理由があるため、情報抽出が非常に困難である。そこで、本研究では、まず HTML を XML に変換することで、仕様に曖昧な部分を解消する。これにより、曖昧さが解消されるばかりか、XML 関連技術を用いることができるようになる。

しかし、これだけでは不十分である。

リスト 1 XML と HTML の比較

例 1 :	例 2 :
<code><info></code>	<code><table border="0"></code>
<code><title>東京工科大学</code>	<code><tr></code>
<code></title></code>	<code><td></code>
<code><url>http://www.teu.</code>	東京工科大学 <code>http://www.teu.ac.jp</code>
<code>ac.jp</url></code>	<code></td></code>
<code></info></code>	<code></tr></code>
	<code></table></code>

リスト 1 は、東京工科大学というタイトル文字列と URL の関連を示す XML と HTML の比較である。例 1 における XML の構造では、情報はセマンティクスによって整理され、各情報が簡素に XPath で記述することが可能である。

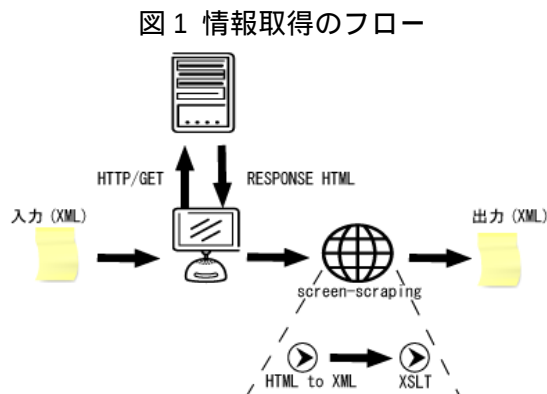
対して、HTML により表される例 2 では、タイトルと URL の文字列に対するセマンティクスが与えられておらず、また各タグの意味もレイアウトのための情報であり、特定の情報を取得するに適した構造ではない。

このため、HTML を単に XML に変換しただけでは、不十分であり、これを変換し情報を整理する必要がある。本研究では、このプロセスに

XSLT[1]を用いることでこれを解決した。

3.2 情報取得のフロー

各サービスにアクセスし、情報を取得、抽出するためのフローは以下の図に示すとおりである。

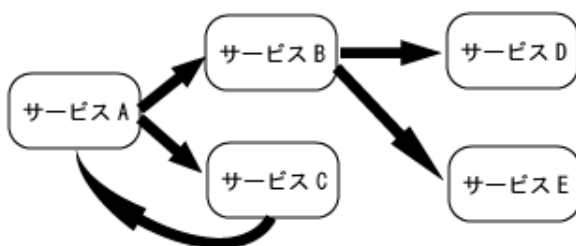


サービスにアクセスし、情報を取得、スクリーン・スクレイピングを行い、情報を抽出し、場合によっては次のサービスのリクエストに用いる。このサイクルを連結することにより、一連の操作を自動化し、最終的に Google Maps に出力する。

3.3 言語の開発

本研究において開発する言語は、3.2 に示したフローを記述し、スクリーン・スクレイピングを行い、整形したデータを出力するためのものである。本研究では、この言語を、XSLT の構造を参考に XML ベースで作成した。

図 2 複雑な連結



本言語では、上に示すような複雑なフロー、および複数の結果を持つようなものに対応するため、言語内に変数を定義し、この変数に情報抽出結果を代入し、変数と変数をつなぎ合わせることで対応した。変数を示す variable タグの中に、その変数を示す一連の操作を記述し、Java プログラムからその変数を示す操作を実行、取得するという流れである。

リスト 2 言語の一例

```
<?xml version="1.0" encoding="Shift_JIS"?>
<ms:mashup
  xmlns:ms="http://www.tzkdev.net/2007/mashup">
  <ms:variable name="result">
    <result>
      <ms:variable name="request_result">
        <ms:request url="http://..."/>
      </ms:variable>
      <ms:transform source="$request_result">
        <xsl:stylesheet...>
          .....
        </xsl:stylesheet>
      </ms:transform>
    </result>
  </ms:variable>
</ms:mashup>
```

本言語にて定義されるタグには、名前空間を用いてプリフィクスを付加することで、通常のタグや、XSLT のタグと区別する。variable タグ以外のほとんどのタグは、実行結果が文字列（この場合 XML）に置換されるタグであり、これが variable に代入されるという寸法である。複数の variable タグを作成、名前を与え、これを連結することでフローの制御を行う。

4. 評価

マッシュアップ用言語を作成し、これを用いて東京都防災マップを構築した。

- (1) 言語実行環境を構築した。
- (2) マッシュアップ可能な HTML と困難な HTML 構造を比較抽出した。
- (3) Google Map を用いて実際にマップを生成した。

(1)では、本言語が正しく動作することを確認した。(2)では、JavaScript や、HTML では表されないデータ構造(例えばカンマ区切り)などに着目し、これらのデータも収集対象にするための検討を行った。(3)では、抽出できたデータを Google Maps の API を用いてマップ化できたことを確認した。

5. 終わりに

本研究により、東京都防災マップを構築することができたが、同時に、情報の抽出が不可能で断念せざるを得ないものも存在した。今後は、さらに多くの避難施設や防災情報を対象にできるようにしていきたいと考えている。

[1]W3C, “XSL Transformations(XSLT) Version 1.0”, <http://www.w3.org/TR/xslt>