

# 気象庁 XML データの活用

江越航\*

## 概要

気象庁では、気象情報のより一層の利活用を図るため、XML フォーマット形式の電文データ配信試行を開始している。これは、気象情報の多岐にわたる可能性を試行し、利活用を促進することをめざして実施されているものである。本稿ではこのデータの受信環境の構築方法と、その後の展開について報告する。

### 1. はじめに

気象は大気中に起こる現象を扱う自然科学の一分野であるとともに、毎日の生活に非常に密着した現象であり、科学館としても取り組んでいきたいテーマである。科学館における気象に関する普及活動の状況においては、別報にて記載した通りである。この気象の普及活動をさらに進めるために、気象庁 XML データの活用を考えた。

気象庁では平成 24 年 12 月より、気象情報のより一層の利活用を図るため、XML フォーマット形式の電文データ配信試行を開始している。これは、従来行われている気象業務支援センター経由の即時情報とは異なるもので、気象情報の多岐にわたる可能性を試行し、より一層の利活用促進を目的としている。

本稿ではこのデータの受信環境構築、および応用の一例として、Twitter との連携について述べる。

### 2. Web 更新情報の入手

気象庁 XML データの配信は、パブサブハブ (PubSubHubbub) と呼ばれるインターネットのプロトコルを利用して配信されている。

現在 web の更新情報を確認するために、RSS (Rich Site Summary / RDF Site Summary) が広く用いられている。RSS とは、見出しと内容情報をメタデータ化と言う形でまとめ、これらの情報を XML ベースの言語で記述したものである。これを、Web サイト上に RSS 文章として記載しておく。

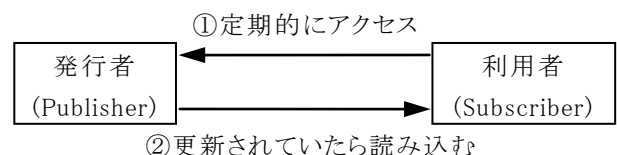
ニュースサイトやブログでは、HTML とは別に RSS 文章を配布している。RSS リーダーによって、この RSS 文章を参照することで、更新のチェックを行うことができる。

また、RSS と同様の技術として Atom と呼ばれるものがある。これは、RSS に複数の互換性のないバージョンが並立していたり、仕様が複雑であったりすることから、まったく新しいフォーマットとして考案されたものである。

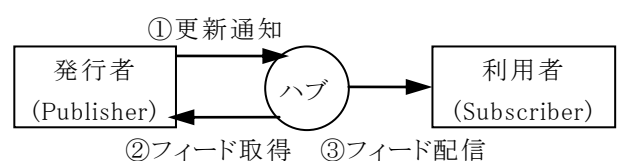
RSS/Atom リーダーは、定期的に更新があるかを問い合わせ、更新があった場合に情報を受け取る仕組みである。そのため、更新情報を受け取るまでにタイムラグが発生するほか、更新の有無にかかわらず問い合わせを行うため、ネットワークやサーバーのリソースを消費する問題がある。

これを解決するための RSS/Atom を拡張した新プロ

#### RSS/Atom



#### PubSubHubbub



\*大阪市立科学館学芸グループ  
e-mail: egoshi@sci-museum.jp

図1 PubSubHubbub の仕組み

トコルとして、PubSubHubbub がある。このプロトコルは、発行者と受信者の間に Hub を介在させることで、更新情報をほぼリアルタイムで受信することを可能にするものである。

発行者が PubSubHubbub に対応したサーバーで更新を行うと、更新通知を Hub に publish する。Hub は更新通知を受けて、サーバーよりフィードを取得し、利用者に更新フィードを送る。これにより、利用者はほぼリアルタイムで更新情報を受信することができる。

### 3. データ受信環境の構築

Hub から送られてくる気象庁 XML データを受信するためには、Hub からの http による GET または POST を処理する必要がある。このため、受信のためのサーバー環境を用意し、フィードの受信プログラムを作る必要がある。

サーバーについては、科学館において現在各学芸員が作成しているホームページ用のサーバーを使用することとした。このサーバー上に php を利用した受信プログラムを作成する。

次の php プログラムは、気象庁 XML 利活用セミナーで紹介されていたサンプルである。

```
<?php
(省略)
$method = $_SERVER['REQUEST_METHOD'];
if ($method == 'POST') {
    $string = file_get_contents("php://input");
    $fp = fopen(date('YmdHis') . "_atom" . ".xml",
"w");
    fwrite($fp, $string);
    fclose($fp);
}
?>
```

Hub から post されたフィードを受信すると、これをファイルとして保存する php プログラムである。

これで気象庁からのデータを受信する前に、本当に受信が可能か、次の手順で php プログラムのテストを行う。

#### 前準備

発行者の役目を果たすための、Atom を発行することができるブログを準備する。今回は、Google のレンタルブログサービス Blogger を利用した。

#### テスト用 Hub の登録

テスト用 Hub として、Google が GAE(Google App Engine)上に立ち上げているものを利用する。これは

<http://pubsubhubbub.appspot.com/subscribe> から利用できる。このページにアクセスすると、図2のような画面が現れる。

図2 テスト用 Hub 画面

このページの Subscribe/Unsubscribe の項目から必要事項を登録する。Callback URL の欄に作成した php プログラムのアドレスを、Topic URL の欄に前準備で用意したテスト用のブログの URL を入力する。Verify type を Synchronous、Mode を Subscribe として、「Do It!」ボタンを押すと、問題なければ特に画面に変化は現れない。

登録が正しく行われたかどうかは、下の Subscriber Diagnostics の項目から調べる。Callback URL の欄に作成した php プログラムのアドレスを、Topic URL の欄にテスト用のブログの URL を入力して、「Get Info」ボタンを押して現れる画面で、State が verified となっていれば、正しく登録されていることを示す。

#### フィードの発行

前準備で作成したブログを更新する。ただし PubSubHubbub に対応していないブログでは、更新情報が Hub に伝わらないので、次の Hub への publish 用ページにアクセスする。

<http://pubsubhubbub.appspot.com/publish>

図3 テスト用 Publish 画面

Publish の項目の Topic URL にテスト用のブログの URL を入力し、「Publish」のボタンを押す。

すると Hub に更新情報が伝わり、さらに Hub から

Subscriber である php プログラムに更新情報が伝わる。この結果、今回の場合はサーバー上に受信したフィードがファイルとして保存される。サーバー上に保存ファイルが作成されることで、php プログラムが動作していることが確認される。

#### 4. 気象庁への登録

前項の内容により php プログラムの動作が確認されたら、気象庁に Hub への登録を依頼する。現在は試行の段階につき、Hub へのユーザ登録は気象庁側で行うため、subscriber の URL 等とともに登録申請を行った。

申請の際、登録を希望する Atom フィードとして、

定時: 天気概況など定時発表されるもの

随時: 警報・注意報など随時発表されるもの

地震火山: 地震・火山に関する情報

その他: 紫外線観測データ、生物季節観測報告等を選択することになっている。今回は定時のみ登録した。

登録申請を行うと、数日後に登録完了メールが届き、Hub よりフィードが送信されてくるようになった。

#### 5. 気象庁からの配信

気象庁からは、随時、以下のような XML 文章で更新情報のフィードが配信されてくる。

```
<entry>
<title>府県天気予報</title>
<id>urn:uuid:*****</id>
<updated>2015-04-13T01:35:24Z</updated>
<author>
<name>大阪管区気象台</name>
</author>
<link href="http://xml.kishou.go.jp/data/***.xml"
type="application/xml"/>
<content type="text">【大阪府府県天気予報】
</content>
</entry>
```

具体的な情報については、XML 電文中に示されているリンク先に保存されているので、必要な情報をさらにリンク先から取得することになる。公開 XML 電文の保存期間は 24 時間のため、必要な情報は 24 時間以内に取得する必要がある。

#### 6. XML 文章の利用

XML 文書は、開始タグに対応した終了タグがあり、その間にデータが記載されている形式になっている。開始タグと終了タグは、入れ子になった構造をしている場合もある。つまり XML 文書は、タグとその間のデー

タを節(ノード)としたツリー構造と考えることができる。

php プログラムにおいて、このツリーにアクセスするには、SimpleXML 関数を使用することができる。例えば、前項の気象庁からの配信 XML 文章が変数 \$string に保存してある場合、

```
$feed = simplexml_load_string($string);
```

```
$name = $feed->entry->author->name;
```

```
$content = $feed->entry->content;
```

とすれば、\$name に“大阪管区気象台”、\$content に“【大阪府府県天気予報】”の文字列を取り出すことができる。このように XML 文章をパース(解析)することによって、XML 文書内の必要な部分を検索することができ、さまざまな活用が可能になる。

#### 7. Twitter への配信

気象庁 XML データ活用の 1 つとして、Twitter への配信を検討した。気象庁 XML データ、Twitter とも即時性が特徴と言えるため、相性がよいと考えられるからである。php プログラムから Twitter へは、以下の手順により配信できる。

##### 別サーバーへのデータ送信

気象庁 XML データを受信している各学芸員紹介用ホームページサーバーは、twitter 投稿に必要な php のコマンドがインストールされていないことから、受け取ったデータのうち、必要なものを別サーバーに転送した。転送は php による post 送信を利用して

```
file_get_contents
```

```
($url, false, stream_context_create($options));
```

の命令を実行することで行った。

##### TwitterAPI の登録

準備として配信用の気象庁 XML Twitter アカウントを作成した。php プログラムからこのアカウントにツイートするには、twitterAPI を使用する。このために、Twitter の Application Management のページから、必要な Twitter アプリケーションの登録を行う。この結果、認証情報となる 4 つのキーを作成することができる。

##### Twitter への投稿

以上の準備を経て、php プログラムより作成した Twitter アカウントに投稿することが可能になる。

実際の投稿のために php プログラム中では、TwitterOAuth と呼ばれるライブラリを使用した。上記で作成した 4 つの認証キーとともに、つぶやく内容のデータをまとめてライブラリを呼び出すことで、Twitter に投稿が行われる。

気象庁 XML データは、各地域の種々のデータが送られてくるが、XML 文章をパースすることで、例えば大阪の天気概況のみ Twitter でつぶやくことも可能になる(なお、今回作成したアカウントはテスト用につき、非

公開設定にしている)。



図4 Twitter 投稿例

## 8. 展示への活用検討

展示場では現在気象コーナーとして、全天カメラで撮影した科学館上空の雲の様子や、国土交通省の X バンド MP レーダーによる雨量情報を表示している。ここに、天気概況として気象庁 XML の電文データからの

情報を表示すると、より興味を持って見てもらえるのではないかと考えている。

ただし展示の構築には、ネットワーク環境の整備と共に、気象庁 XML データの利用指針に基づき、気象業務法に留意した内容にする必要がある。

## 9. おわりに

以上、主に気象庁 XML データの受信環境構築、および応用の一例として、Twitter との連携について述べた。このデータを利用することで、種々の新たな気象情報の活用が考えられる。

またこのデータには、通常の天気予報や防災情報に加え、植物の開花日や鳥の初鳴きなど、生物季節観測のデータも含まれている。このようなデータを蓄積することで、広い範囲での季節の違いを感じる仕組みができる可能性もある。

現在、データ受信環境が整ったことから、今後データの加工方法を確立し、さらに活用していきたいと考えている。

なお、本研究は平成 26 年度科学研究費補助金(奨励研究)の助成を受けて行なったものである。