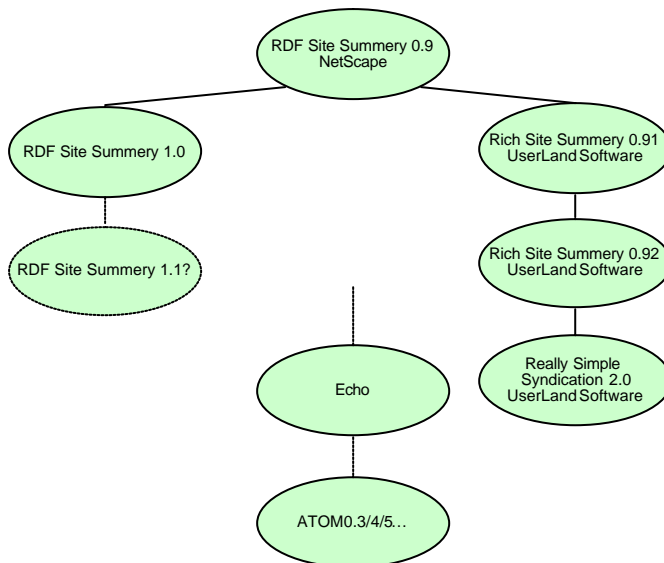


Atomの概要と オープンソースライブラリを使った iPlatプロジェクトでの実装

Sky株式会社 玉川竜司
(tamagawa@skygroup.jp)

2005/7/5

Atomに至る歴史





Atomを使うと何がうれしい?



- RSSの進化は、基本的にはもう止まっている
 - RSSはReally Simpleであるがゆえに成功した
 - ただし、より複雑なデータを扱うための厳密な仕様定義という点が弱い
 - RSS1/2系の仕様は、その策定者たちによって「凍結」が宣言されている(マイナーなメンテナンスはある?)
- Atomでは、仕様に関する議論がオープンかつ活発に行われている
 - やり取りされるデータのフォーマットや、そのエンコード/エスケープの仕方が厳密に定義されている
 - よりリッチな情報を、互換性の心配なくやりとりできるようになる(だろうと思われる)
- 乱立するコンテンツ操作のAPIが統一される可能性がある
 - xmlrpcをベースにする一連のAPIは一定の成功を収めているものの、複数のAPIが乱立している
 - Atom Publishing Protocolによって統一されれば、ソフトを書くほうはもちろん楽



Atom Syntax – 簡単な例



```
<?xml version="1.0" encoding="UTF-8"?>
<feed version="draft-ietf-atompub-format-05:do not deploy"
  xmlns="http://purl.org/atom/ns#draft-ietf-atompub-format-05">
  <head>
    <title>Example Feed</title>
    <link href="http://example.org"/>
    <updated>2003-12-13T18:30:02Z</updated>
    <author>
      <name>John Doe</name>
    </author>
  </head>
  <entry>
    <title>Atom-Powered Robots Run Amok</title>
    <link href="http://example.org/2003/12/13/atom03"/>
    <id>vemmi://example.org/2003/32397</id>
    <updated>2003-12-13T18:30:02Z</updated>
  </entry>
</feed>
```



Atom Syntax – 特徴



- ConstructとElementからなる厳密な定義
- AtomのデータはElementからなり、Elementの形式はConstructで規定される
- TEXT ConstructのTypeはTEXT/HTML/XHTMLのいずれか。それぞれのタイプが指定されている場合に、たとえば<をどう扱うかは厳密に規定されている
- ただし、Content Elementの内容は、上記の3タイプ以外にMIMEのMedia Typeを指定することができる
- RDFベースではない
- 名前空間を使った拡張が可能



Atom Publishing protocol (1)



- コンテンツを編集するためのプロトコル
- HTTPとXMLを使用する
 - ただし、xmlrpcやSOAPとは大きく異なる (REST)。
 - HTTPで用意されているメソッドをそのまま編集機能に割り当て、操作の結果もそのままHTTPのレスポンス・コードで返す
 - 密な連携には向かない。
 - XMLのドキュメントはAtom Syntaxを使う。
 - 操作のためのURIは、やり取りされるAtom Syntaxに基づくドキュメント中に記述される
- 認証についての規定がある
 - やり取りされるコンテンツを盗聴から保護するものではない
 - コンテンツの編集権を保護するためのもの



Atom publishing protocol (2)



- 4種類の操作—コンテンツの取得・作成・編集・削除
 - 取得 = GET
 - 作成 = POST
 - 編集 = PUT
 - 削除 = DELETE
- 4種類のURI—PostURI・EditURI・FeedURI・ResourcePostURI
 - PostURI
 - エントリを作成するためのURI
 - POSTが必須
 - EditURI
 - エントリを編集するためのURI
 - PUT・GET・DELETEが必須
 - FeedURI
 - エントリを取得するためのURI
 - GETが必須
 - ResourcePostURI
 - エントリでないリソースを作成するためのURI
 - MIMEタイプつきのリソースを直接Postできる



その他の話題



- Atom Notification Protocol
 - コンテンツの更新情報を通知するためのプロトコル
 - まだ話が始まったばかり？
- 普及の状況
 - CMSはプラグインで対応しているケースが多い
 - メジャーなblogサーバはこれから
 - ライブラリの多くは対応してきている
 - blogサーバとクライアントツール=鶏と卵？



オープンソース・ライブラリ



- 普及には手軽に使えるライブラリが必須！
- JavaとPHPについて調べてみました
- Javaのライブラリリスト
 - <http://java-source.net/open-source/rss-rdf-tools>
- Jakarta FeedParser
 - <http://jakarta.apache.org/commons/sandbox/feedparser/>
- **MagpieRSS: RSS for PHP**
 - <http://magpierss.sourceforge.net/>
 - PHPベースのパーサ
- FeedCreator.class.php
 - <http://www.bitfolge.de/rsscreator-en.html>
 - RSS/Atom Feedの生成ライブラリ



オープンソース・ライブラリ (2)



- たいいていちゃんと使える (Java/PHP自体はUTF8で問題なし)が、ライブラリ自体が使用している文字のエンコードについては要注意
- エンコードがISO-8859固定になっていたり (FeedCreator)、8bit目が一切通らなかったり (Apache XMLRPC)といったことがあるので、気をつけましょう
- ライセンスの処理にももちろん注意が必要
- Atom仕様の更新にはなかなかついて行くのが大変そう・・・0.3ベースのものが多い？

The screenshot displays the iPlat project's user interface. At the top, it says "iPlatプロジェクト:UI" and features the XML Consortium logo. The main window shows a map with several red markers. A callout box points to one of these markers, stating: "iPlatBlogの記事,トラックバック先の記事が表示対象となる". Below the map, there are several smaller windows: "SKY ACTIVE FIELD" (with a note: "GPS携帯で投稿した時点で,すぐにその情報を閲覧できる"), "他のBlog" (with a note: "ブログに投稿した時点で,すぐにその情報を閲覧できる"), and "iPlatBlog". The interface includes a sidebar with various icons and a top navigation bar.

Copyright © XMLコンソーシアム 2005 All rights reserved. 11

iPlatプロジェクトでのターゲット

- WebServicesに基づく全体システム (TravelPlanner) に対して、緩やかな連携に基づく情報提供を行う
- 既存の情報インフラの中で流通している情報を有効活用し、メタデータを使って連携させる
- WebServicesとRSS/Atom Feedという二つの世界をつなぐ:Blog情報集約Webサービス
 - キーワードと位置情報を元に検索を行い、ヒットした結果をAtomで返す
 - 返されたAtom中にはコンテンツのURIがある

Copyright © XMLコンソーシアム 2005 All rights reserved. 12



連携に参加するシステム



XML Consortium

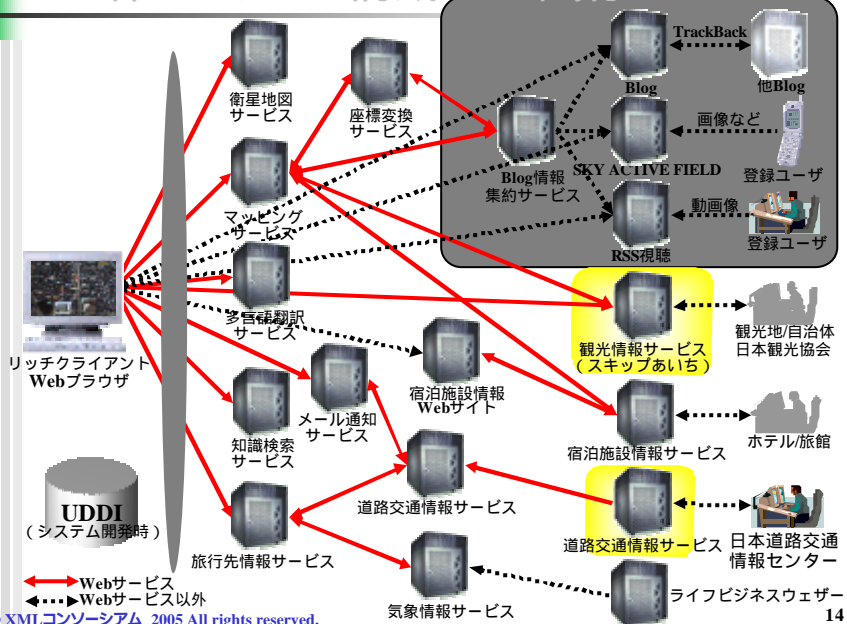
- Blog (Nucleus)
 - オープンソースのBlog/CMSシステム。
 - PHP/MySQL上で稼動する。
- SKY ACTIVE FIELD (Sky 株)
 - 携帯からの画像つきレポートを管理するシステム。
 - GPS機能付き携帯を使うことで、位置情報を持ったレポートが扱える。
- RSS視聴
 - ブラウザ上で稼動する、高機能動画ビューア。
 - 内部情報の管理にRSSを用い、見たい部分へのアクセスを容易にする。



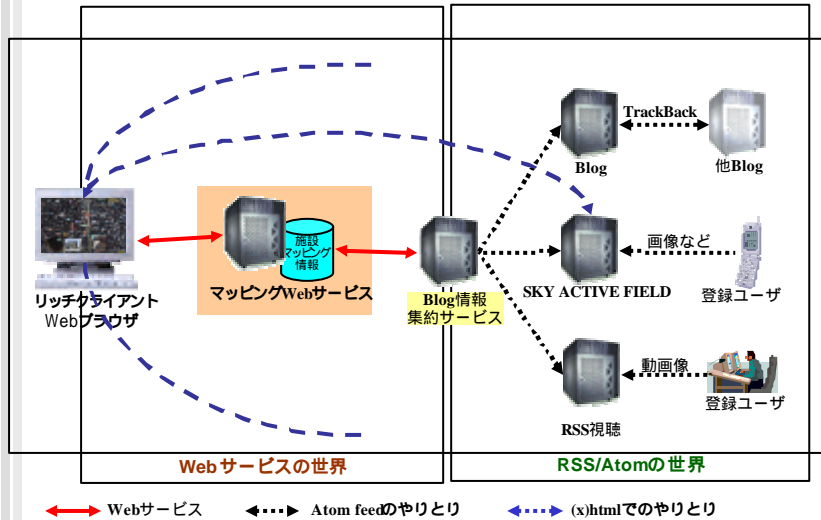
全体システム構成との関係



XML Consortium



二つの世界をつなぐ :情報の流れ



AtomFeedの実装:ツールの選定



- PHPベースのAtom生成ライブラリ:FeedCreator
 - <http://www.bitfolge.de/rsscreator-en.html>
 - 使い方は簡単 :
 - ```
$rss = new UniversalFeedCreator();
$rss->useCached();
$rss->title = "PHP news";
$rss->description = "daily news from the PHP scripting world";
$rss->link = "http://www.dailyphp.net/news";
$rss->syndicationURL = "http://www.dailyphp.net/".$PHP_SELF;
(MySQLの検索)
while ($data = mysql_fetch_object($res)) {
 $item = new FeedItem();
 $item->title = $data->title;
 $item->link = $data->url;
 $item->description = $data->short;
 $item->date = $data->newsdate;
 $item->source = "http://www.dailyphp.net";
 $item->author = "John Doe";

 $rss->addItem($item);
}

$rss->saveFeed("ATOM", "news/feed.atom");
```





# 生成されるAtom



XML Consortium

```

<?xml version="1.0" encoding="utf-8"?>
<!-- generator="FeedCreator 1.7.2" -->
<feed version="0.3" xmlns="http://uri.org/atom/ns#" xmlns:iPlat="http://www.xmlconsortium.org/iPlat">
 <title>Nucleus Atom Feed</title>
 <tagline>Nucleus article and Trackback search for iPlat</tagline>
 <link rel="alternate" type="text/html" href="http://iPlat/">
 <id>http://iPlat/</id>
 <modified>2005-07-04T19:34:24+01:00</modified>
 <generator>FeedCreator 1.7.2</generator>
 <entry>
 <title>%E5%90%8D%E5%8F%A4%E5%B1%8B%E5%9F%8E%E3%81%AB%E3%81%A4%E3%81%84%E3%81%A6</title>
 <link rel="alternate" type="text/html" href="http://iPlat.xmlconsortium.org/nucleus/index.php?itemid=3"/>
 <created>1970-01-01T00:00:00+01:00</created>
 <issued>1970-01-01T00:00:00+01:00</issued>
 <modified>1970-01-01T00:00:00+01:00</modified>
 <id>http://iPlat.xmlconsortium.org/nucleus/index.php?itemid=3</id>
 <author>
 <name>Sky株式会社</name>
 </author>
 <summary>%E5%90%8D%E5%8F%A4%E5%B1%8B%E5%9F%8E%E3%81%AB%E3%81%A4%E3%81%84%E3%81%A6%E
3%81%AE%E6%83%85%E5%A0%B1%E3%81%AF%E3%81%93%E3%81%A1%E3%82%89%E3%82%92%E3%81%94%E8%
A6%A7%E3%81%8F%E3%81%A0%E3%81%95%E3%81%84%E3%80%82</summary>
 <iPlat:keyword>名古屋城</iPlat:keyword>
 </entry>
</feed>

```

- 外部「から来る可能性のあるタイトル・サマリーは安全のためurlencode
- 名前空間iPlatの導入による拡張 :キーワードと位置情報



# FeedCreatorを試ってみて



XML Consortium

- 使用法は非常に簡単
  - もちろんテキストデータとしてAtomを書き出しても手間は知れているが...
  - それでもやはり簡単。他のフォーマット(RSSx.x)への書き出しも可能
- そのままでは日本語の使用には問題あり
  - エンコードがISO-8859に固定されている
  - ソースを一行変更することで問題なくUTF8が使えるようになった(ように見える)
- ちなみにライセンスはLGPL



## Resources - web



- Atom Wiki (本家)
  - <http://www.intertwingly.net/wiki/pie/FrontPage>
- AtomEnabled
  - <http://www.atomenabled.org/>
- The Web Kanzaki (日本語)
  - <http://www.kanzaki.com/docs/sw/atom.html>
- The Atom Syndication Format
  - <http://www.ietf.org/internet-drafts/draft-ietf-atompub-format-09.txt>
- The Atom Publishing Protocol
  - <http://www.ietf.org/internet-drafts/draft-ietf-atompub-protocol-04.txt>



## Resources – Mailing List



- **atom-syntax mailing list**
  - Atomのフォーマットに関する議論。非常に活発
  - 現在Atom0.9に関する議論が進行中
  - <http://www.imc.org/atom-syntax/index.html>
  - 1/17-7/1で3945通。平均約21通/日
  - ちなみに<http://ietf.levkowitz.com/drafts/atompub/format/>で変遷を見ることができます
  - FeedValidator
    - <http://www.atomenabled.org/developers/syndication/validator.php>
- **atom-protocol mailing list**
  - Atom APIに関する議論。閑散としていましたが、徐々に活発に
  - <http://www.imc.org/atom-protocol/index.html>
  - 1/17-7/2で741通



これで終わりです



- ご静聴ありがとうございました。