

Patrick Patterson
L597 – 1 credit Independent Study
December 9, 2005

A look at the use and Creation of RSS/Atom Feeds

This research is a continuation of my internship at the Monroe County Public Library (MCPL) that took place during the summer of 2005. During this time, I worked in the Information Systems department under the guidance of Paula Gray-Overtoom looking into how RSS Feeds could be incorporated into the libraries web site. The libraries website has approximately 500 html pages making it difficult for patrons to find the information they are looking for. It was determined that developing RSS Feeds will help provide a better service to library patrons.

What is RSS?

RSS is used to syndicate content over the Internet through the use of a news aggregator. Syndication occurs when a website makes some of their content available to other sites at no cost. A good non web example is the Associate Press who makes their content available to people around the world.

RSS feeds can be used by a library to provide a more personalized service to patrons. Libraries are creating feeds to inform patrons about current events and when new books arrive that may be of interest to them. A feed makes it easier for people to keep informed about events that take place at their local library. For example, we have created a feed for the events that take place at the MCPL from the current day until seven days in the future.

Feeds are written using the XML programming language to describe content of a website that is updated on a regular basis. This gives people the opportunity to easily keep track of several different websites through the use of a news aggregator. A news

aggregator is a computer application that allows a person to subscribe to an RSS Feed and keep track of several different ones at the same time. There are two different kinds of news aggregators: web based and desktop. Web based aggregators are online portals that check RSS feeds for updates on a regular bases and are accessible from anywhere provided a person has access to the internet. A desktop aggregator is a piece of software that allows a person to manage syndicated content they would like to receive. For example, a person can set up the application to display content in multiple ways and search for content based upon a time interval of their choice. Feeds have been designed to integrate themselves into a website that uses a content management system, such as a web blog. As a person updates their blog the information is saved into a database including a title, date, and URL of where the page is located (i.e. <http://www.mysite.com/news/oct>).

The following standards RSS 1.0, RSS 2.0, and Atom are used to syndicate content over the web. The 2.0 standard will be discussed first, because of its simplicity compared to the other formats. The RSS 1.0 standard uses the Rich Description Framework (RDF) to describe a feed in more detail. Atom is relatively new, but works and acts just like the other two standards. It is important to understand RSS 1.0 and 2.0 are completely different from each other and that 2.0 is not a newer version of 1.0.

How are feeds being used?

Many news organizations are creating feeds and making them available for free to encourage people to incorporate them into their website. This has become another way for news organizations to increase their visibility on the web. According to Mofit (2003), “It is extremely easy for an end user to monitor literally hundreds of websites using an

RSS aggregator but is very time consuming to monitor the same sites without one”
(p.129). A news aggregator is a software application that allows a person to subscribe to a feed and uses an XML parser to display the content of a feed in human readable format. This can be done by using a free web-based service or downloading a desktop reader and installing it on your computer.

A feed makes it easy for a person to find out when the content of their favorite website(s) have been updated. We can look at the following fictional scenario to better understand how a person can benefit from subscribing to several different feeds. A climatologist is doing research on the current weather trends in the United States to help predict when and where the next hurricane may occur. This person may use up to forty different weather related sites to keep track of current weather patterns making it very time consuming to sort through the most current data. A news aggregator will allow the climatologist to search all forty websites at one time and provide him or her with a brief summary of the content that has been recently updated. Thus improving the time and effort it will take to go to each website and search for the most recent information.

Libraries are using feeds to deliver updated content to patrons, such as new materials and job related information. For example, the Kansas City Public Library uses RSS feeds to let a patron know when their website has been updated. They use a content management system attached to a database making it easy to create a feed that will automatically update itself when a person updates the content. At the moment they are providing information about the aftermath of Hurricane Katrina that hit the Gulf Coast region. The home page briefly describes the event and when a user clicks on the hyper-link they are redirected to a page that has more detailed information about the tragedy. A

RSS feed has been created for this page allowing a person to use a news aggregator to subscribe to the feed.

A RSS feed can be written using a web-based editor to convert the content of a web site into the XML required by a news reader. This paper will explore the elements that are required to write the XML by hand to create a feed using the RSS 1.0, RSS 2.0, and Atom standards. Please be aware that the 1.0 and 2.0 standards are completely different from each other. RSS 1.0 uses the W3C's recommendation for metadata and is considered to be very flexible and more complex than other two versions. On the other hand the 2.0 standard was created to be very easy to use by people who have limited knowledge of metadata and XML.

Brief History

Netscape has been recognized by many as the company that created RSS and in 1999 they released My Netscape Network that allowed people to view web content by going to one location (Afzali, 1999, para. 1). This helped Netscape attract companies that were looking to advertise on the Internet for the first time, because they could market themselves to a captive audience. My Netscape was created as a web portal that allowed a person to create a personalized start page based upon their preferences. The system was based on the RDF that was initiated by Ramanathan V. Guha to describe web related content about abstract concepts (a more detailed description is provided later).

The two key people in the development of RSS are Dan Libby who worked for Netscape and Dave Winer who was employed by Userland. Mr. Libby created RSS 0.9 which was a fully functional RDF System that Netscape felt was too complicated for users and as a result he made changes to the system to create a watered down version (0.91)

that had limited RDF capabilities. At the same time Dave Winer was working on his own 0.91 version that eliminated RDF completely and was released a few days after Mr. Libby's version. Having two different 0.91 RSS standards caused confusion and a major rift amongst the web community. When Userland created an updated version they named it 2.0 to eliminate confusion. Netscape eventually abandoned the work they were doing and Rael Dornfest of O'Reilly continued working with the originally 0.9 version and later released the 1.0 standard. This has resulted in a heated debate over which RSS standard should be used. On one side you have people who believe it should be as simple as possible to use and others who feel that you must use RDF to truly describe your content. This has led to a third group of individuals who have been working towards a universal standard called Atom.

RSS 2.0

This version stands for Really Simple Syndication which is based on simplicity and is an outgrowth of David Winer's work at Userland. The RSS Specifications and RSS Feeds website states that "On July 15, 2003, Userland Software transferred ownership of its RSS 2.0 specification to the Berkman Center for Internet & Society at Harvard Law School [under the Creative Commons License]" (RSS History, para. 3). The specifications for the RSS 2.0 standard can be found at Harvard Law's RSS Website located at the following URL <http://blogs.law.harvard.edu/tech/rss>.

The first step in creating a feed is to begin with an XML declaration which indicates the version that will be used (`<?xml version="1.0" encoding="utf-8"?>`). Every XML document needs to have a root element that serves as a container for all other items. The root element for the 2.0 standard is `<rss version="2.0">` and contains the **channel** sub

element which is made up of several elements used to describe the feed. It requires the following items:

- title – the name of the feed you are creating.
- link – A URL that points to the corresponding HTML website listed in the channel element (i.e. <http://www.heraldtimesonline.com/sports>).
- description – Words written in plain text to describe the feed

In addition to the required elements there are 16 optional sub elements that can be included, such as copyright date, published date, and e-mail address of the webmaster (please consult the RSS 2.0 Specifications for a complete listing). Most of the content of the 16 optional sub elements are static and are a good way to add additional content to describe your feed. Inside the channel sub-element you are required to have at least one sub-element **item** that is used to describe the feed. The simplest required item would look like:

```
<item>  
<description>This is the required element for the sub-element item.</description>  
</item>
```

There are ten sub elements that can be used to describe the **item** in more detail with **title** being the only one required. The other most useful sub elements for an item are:

- link – The URL of the story.
- author – Contains the e-mail address of the person who created the feed.
- pubDate – The date the item was published in RFC 822 format (Hammersly, 2005, p. 32). For example `<pubDate>Sun, 11 2005 00:13:02 GMT</pubDate>`

Consider the use of an RSS feed for the Herald Times Newspaper in Bloomington, Indiana. In this scenario each of the different sections of the paper: front page, sports, entertainment and classifieds will become a **channel** to create four different feeds to syndicate content to readers. Within the sports section there are several different stories which will become **item** sub-elements. For example, a simple feed for the sports section of the Herald times might look like:

```
<?xml version="1.0" ?>
<rss version="2.0">
  <channel>
    <title>Herald Times – Sports</title>
    <link>http://www.heraldrtimesonline.com/sports.html</link>
    <description>This feed contains the most up to date sports news in
    Bloomington Indiana</description>

    <item>
      <title>IU Men’s Soccer</title>
      <link>http://www.heraldrtimesonline.com/sports/IU/soccer.html
      </link>
      <description>The Men’s soccer team at Indiana University is
      looking to win there third straight national title.</description>
      <author>patrick@heraldrtimes.com</author>
    </item>

  </channel>
</rss>
```

After creating this file you will need to save it with the **.rss** extension and then validate the XML file to make sure it can be read by a news aggregator. A useful validator for syndicated feeds is the FEED Validator which can be found at <http://feedvalidator.org/>.

RSS 1.0

The standard is referred to as RDF Site Summary originally designed by Dan Libby while working at Netscape. It has been designed to use metadata to describe

resources that are available through the internet, such as author, title, and the date of publication. “RDF is based on the idea of identifying things using Web identifiers and describing resources in terms of simple properties and property values” (W3C, para. 3). This allows an application to process the unique identifier to extract important information from a web page and create links to pieces of information throughout the web. The important thing to keep in mind is that RDF allows a person to create an XML document that can point to any identifiable thing on the web, such as an image or name and not just the URL of a web page.

Rewriting the RSS 2.0 example for the herald times to use the 1.0 standard would look like:

```
<?xml version="1.0" ?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns="http://purl.org/rss/1.0/">

<channel rdf:about="http://heraldtimesonline.com">
    <title>Herald Times – Sports</title>
    <link>http://www.heraldtimesonline.com/sports.html</link>
<description>This feed contains the most up to date sports news in Bloomington
Indiana</description>

<items>
<rdf:Seq>
<rdf:li rdf:resource="http://www.heraldtimesonline.com/sports/IU/soccer.html" />
</rdf:Seq>
</items>

</channel>

<item rdf:about="//heraldtimesonline.con/sports/IU/soccer.html">
    <title>IU Mens Soccer</title>
    <link>http://heraldtimesonline.com/sports/IU/soccer.html</link>
<description>The Mens soccer team at Indiana University is looking to win their third
straight national title.</description>
</item>
</rdf:RDF>
```

This example looks very complicated and intimidating to create; however breaking it down into small chunks of information will make it easier to understand. One way to think of RSS 1.0 is a metadata standard that is used to describe the content of your website that is updated on a regular basis.

The root element of the document `<rdf:RDF: />` is used by an XML parser to describe the other required elements used as part of the 1.0 standard. The first required sub element is `<channel rdf:about="http://.">` that uses metadata to describe itself to a newsreader. It requires the use of a URL that will point to a feed or to the website it represents. The following **channel** sub elements are required:

- title – the title of the feed `<title>Herald Times – Sports</title>`
- link – URL of the site
`<link>http://www.heraldtimesonline.com/sports.html</link>`
- description – a brief description of the feed
`<description>This feed contains...</description>`
- items – They consist of two sub elements `<rdf:Seq>` and `<rdf:li resource="http: " />` to create an RDF relationship between the **channel** and each **item** of the document.

Please keep in mind that *item* is plural and is a sub element of the RDF and that *item(s)* is a sub element of the channel. This is very important to understand when trying to understand why your feed will not validate.

The second required sub element `<item ref:about="http://. ">` is the most important piece of information, because it is parsed by a news reader to describe the content of your feed. In our example a news reader will display something like:

IU Men's Soccer

The Men's soccer team at Indiana University is looking to win their third straight national title.

<http://heraldtimesonline.com/sports/IU/soccer.html>

Only the title and link sub elements are required, but a short description should be included to describe the feed. After writing a feed using the 1.0 standard you need to save it with the **.rdf** file extension.

Atom

Atom is the new up and comer that began in 2003 when a group of people came together to start working on a new standard to publish and syndicate content. There design philosophy corresponds with the open source community in which Atom is being created through the use of a Wiki where anyone can participate in its development. The Wiki states: "Let's put aside our differences and work together to achieve [our] goals". If you are interested in finding out more information or contributing visit the organizations website at <http://atompub.org/>. The focus is to "give all the information you might need to display the content and the first order of information about the content: who wrote it, and when ..." (Hammersley, 2005, p. 117).

Atom has been designed to do the following:

1. Syndicate content
2. To assist in the retrieval and creation of online resources such as a web blog.

Using the information presented earlier for the Herald Times would produce a simple

Atom feed that would look like:

```
<?xml version="1.0"?>

<feed xmlns="http://www.w3.org/2005/Atom">

<title>Herald Times Sports Feed</title>
<updated>2005-10-02T10:37:00Z</updated>
<link href="http://www.hoosiertimesonline.com"/>
<author>
  <name>Patrick Patterson</name>
</author>
<id>http://hoosiertimesonline.org/2005/123</id>

<entry>
<title>Sports-IU Soccer</title>
<link rel="alternate"
href="http://www.hoosiertimesonline.com/sports.html"/>
<id>http://hoosiertimesonline.org/2005/123</id>
<updated>2005-10-02T10:37:00Z</updated>
<summary>The Men’s soccer team at Indiana University is looking to win their third
straight national
title</summary>

</entry>

</feed>
```

The feed requires the following elements:

- title – the name of the feed
- link – This element makes a connection between the Atom document and its html version. The rel attribute “alternate” is required and is used by metadata to link documents together.
- updated – refers to the last time the content was updated
- author – person who is responsible for the feed
- id – the identification of your feed using a Universal Resource Identifier (URI)

These elements are used to describe the feed and they are equivalent to the Channel element used in the RSS 2.0 format. The **entry sub element** is used to describe an article that is part of a feed, such as a single post in a web log. It requires the same items used to describe the feed with the addition of a summary that is used to briefly describe each entry. Other useful elements that can be used are copyright information and contributor used to list other authors. After creating your XML file it needs to be saved with the file extension **.atom**. At the moment the Feed Validator mentioned earlier is the only option for validating an atom feed.

During the course of a few months Atom has undergone several changes reflecting global participation in creating a new standard. This paper has explored the Atom 1.0 standard that was approved August 2005. Just a few weeks ago a new Internet-draft was conceived proposing a new 1.3 standard which is scheduled to be approved in January.

What do I use?

Making a decision on what specification to use depends on a persons experience with XML and Metadata. At this point in time it does not matter which standard is used, because most news readers are capable of displaying all of the formats discussed. However, it is important to briefly look at the advantages and disadvantages to make a decision on the one that best meets your needs.

The RSS 2.0 standard is very easy to learn, but it is unable to describe a feed in great detail (theoretically making it harder for someone to find a feed). The advantage of this standard is it only uses the required information that is needed to create and view a working RSS feed. This format is limited in scope and is no longer being developed.

Atom's strength is that it is being developed by several people that have the same goal to create a universal standard. Its most distinctive feature is that it is being designed to syndicate content and publish web log content using an API¹. The confusion of the different RSS versions that resulted in political arguments makes Atom more attractive, because it eliminates confusion.

On the other hand RSS 1.0 has been designed as “a lightweight multipurpose extensible metadata description and syndication format” (RDF Site Summary, section 1). The use of Metadata allows a person to provide more details about a website, potentially making it easier to find the information you are looking for. As discussed earlier this standard uses RDF to make a relationship between the items and attributes of a feed. It has been designed to take advantage of the different metadata schemas to describe the content of a website.

After investigating the different standards available to create a feed, I recommend that the Monroe County Public Library use the 2.0 standard because it will be easy for staff to learn how to use. The library's mission is to provide access to information, a place to gather, and support learning. RSS Feeds will help the library to meet these goals by making it easier for people to locate up to the minute information about events that take place at the library. As with any other technology the RSS 1.0, 2.0 and Atom feeds will change over time providing people to think about new and interesting ways to deliver content.

¹ API - Application Program Interface which is a standard set of protocols for developing a software application.

References

- Afzali, C. (1999). *Netscape Launches Publishing Program*. Retrieved September 3, 2005, from http://www.internetnews.com/bus-news/article.php/3_80051.
- Byrne, G. (2005). RSS and Libraries – Fad or the Future? *Feliciter* 51(2), 62-63. Retrieved July 2, 2005 from Academic Search Premier EBSCO database.
- Hammersly, B. (2005). *Developing Feeds with RSS and Atom*. Cambridge: O'Reilly.
- Moffat, M. (2003). RSS – a primer for publishers and content providers. *The New Review of Information Networking* 9(1), 123-144. Retrieved July 15, 2005 from Academic Search Premier EBSCO database.
- RDF Site Summary (RSS) 1.0 (2000) *Home page*. Retrieved September 17, 2005 from <http://web.resource.org/rss/1.0/spec#s5>
- RSS Specifications and RSS Feeds (n.d.) *History of RSS*. Retrieved September 3, 2005 from <http://www.rss-specifications.com/history-rss.htm>

RSS Feed Example

Kansas City Public Library - <http://www.kclibrary.org/index.cfm>

Web Resources

- Atom Specifications - <http://atompub.org/2005/08/17/draft-ietf-atompub-format-11.html>
- BBC Syndication-
<http://news.bbc.co.uk/shared/bsp/hi/services/htmlsyndication/html/default.stm>
- Creative Commons License - <http://creativecommons.org/licenses/by-sa/1.0/>
- RSS 1.0 Specifications - <http://web.resource.org/rss/1.0/spec>
- RSS 2.0 Specifications - <http://blogs.law.harvard.edu/tech/rss>