

Syndicating Your Web Site Content with RSS

by Christopher Siwa

July 2008

Table of Contents

- [Introduction](#)
- [General Information](#)
 - [What is RSS?](#)
 - [How do I read an RSS Feed?](#)
 - [Why should I make a feed available?](#)
 - [Different RSS formats](#)
- [Creating a Static Feed](#)
 - [Choosing content for your feed](#)
 - [Basic XML syntax](#)
 - [All elements must have a closing tag](#)
 - [XML tags are case sensitive](#)
 - [XML tags must be properly nested](#)
 - [Forbidden characters in XML](#)
 - [Creating your feed](#)
 - [Optional elements](#)
- [Creating a Dynamic Feed](#)
 - [Using ASP to dynamically generate your feed](#)
 - [RSS functions used](#)
 - [ApplyXMLFormatting](#)
 - [FormatRSSDate](#)
- [Marketing Your Feed](#)

I. Introduction

Take a moment and think about all of the Web sites that you like to visit on a daily basis. Depending on how many Web sites you consistently visit, this may be tedious. You have to visit each Web site, wait for it to load, then manually sort through all the information and keep track of what you have and have not read.

RSS helps make things easier for you because the information comes to you. Nowadays, more Web sites are including their content in RSS feeds and making them readily available. All you have to do is choose an aggregator to manage and present all of the feeds that you subscribe to.

II. General Information

a. What is RSS?

There are so many different versions of RSS out there, and even though there are many, the term "RSS" can be used to generally refer to any of them. RSS is basically an XML-based format that allows a Web site's content to be syndicated. This XML format may contain a summary of content or full text from a Web site.

b. How do I read an RSS feed?

RSS feeds can be read through the use of special software called an "RSS reader," "feed reader," or an "aggregator." These feeds are formatted in a standard way so that many different aggregators may access them. When you choose an aggregator, choose it based on your preference of:

- How the feed is presented to you – for example, combine several related feeds into a single view, hiding entries that you've already read, or categorizing the feeds you subscribe to.
- How often it checks for an update – once an hour, once every couple of hours, etc.

Finding an aggregator is easy. Aggregators have been integrated into software that most of us use every day. The latest versions of Internet Explorer and Firefox will automatically offer the option to subscribe to a feed if they detect one. There are also free downloadable aggregator add-ons for Microsoft Outlook.

If you prefer a third party aggregator, there are plenty of free ones out there. Some popular examples include Bloglines, Google Reader, My Yahoo! and NewsGator. Once you've selected one, simply copy the feed's URL and paste it in your aggregator.

c. Why should I make a feed available?

RSS feeds make it easier for people to stay up-to-date with a Web site in an automated manner without having to visit that Web site. One may argue that it is a bad idea to not have people visit your Web site because you are just giving the content away.

In reality, RSS helps improve your Web site's visibility. You determine what content you want to place in your feed. If you provide visitors with enough information that will spark interest, you can still drive them to your Web site. Also, RSS helps make it easier for your audience to maintain a strong connection with your Web site because the latest content will always reach them.

For example, the AALL Calendar of Events is a member-driven service so new events may be posted at any time. Without a feed, a member would have to visit the Calendar of Events, sort through the different events, and

determine if they have read them or not. With a feed, they will always receive the latest events posted to the Calendar.

d. Different RSS formats

There are many different versions of RSS, but the most common choices are RSS 1.0 and RSS 2.0. Both versions are XML-based and have the same basic structure, but each version has its benefits and drawbacks. Most aggregators these days can process any feed regardless of format, so selecting a version is a matter of personal taste.

RSS 1.0, which stands for RDF Site Summary, is a version of RSS that incorporates RDF. RDF is a Web standard for metadata. Because RDF is used, any RDF processor can understand RSS without knowing anything about it in particular. As a result, RSS 1.0 is more extensible and useful if you want to integrate it into Semantic Web, but it is not easily authored by hand.

RSS 2.0, which stands for Really Simple Syndication, is a version of RSS that focuses on simplicity. RSS 2.0 is very easy to author by hand, but it is not extensible as RSS 1.0. Because this is the easier version to work with, this is the version that we will be using in our examples.

Finally, there is Atom. RSS 1.0 and RSS 2.0 are not governed by a standard body but rather a small group of people. Because of this, specifications may change at any time. Atom is the Internet Engineering Task Force's (IETF) attempt at a standard syndication format, which in turn would help bring stability. Currently, Atom is not as widely accepted or supported as RSS 1.0 or 2.0 because it is relatively new.

III. Creating a Static Feed

a. Choosing content for your feed

Think about what type of content you would like to include in your feed. While it is a good start to have one "giant feed" that encompasses everything on your Web site, it is best to divide your content into separate feeds. This allows subscribers the freedom to precisely target their interests.

For example, newspaper Web sites that offer RSS feeds will offer a feed for each of their sections – latest news, local news, business, sports, entertainment, etc.

b. Basic XML syntax

Before we create our feed, let us review some basic XML syntax.

i. All elements must have a closing tag:

In HTML, you do not need to close all of your tags.

Example:

```
<p>This is an example sentence.
```

In XML, everything must have a closing tag.

```
<p>This is an example sentence.</p>
```

ii. XML tags are case sensitive:

```
<item>This is incorrect</Item>
```

```
<item>This is correct</item>
```

iii. XML tags must be properly nested:

In HTML, this is okay.

```
<b><i>This sentence is bold and it is italicized.</b></i>
```

In XML, all tags must be properly nested within each other. For example, because `<i>` was opened within ``, it must be closed `</i>` before you close ``.

```
<b><i>This sentence is bold and it is italicized.</i></b>
```

iv. Forbidden characters in XML:

XML files cannot contain certain characters, and they must be replaced with their HTML equivalent.

Characters that cannot be used	Description	Replace with
<	Less than	<
>	Greater than	>
&	Ampersand	&
'	Single Quote	'
"	Double Quote	"

For example:

Incorrect format:

```
<b>This is my first feed</b>
```

Correct format:

```
&lt;b&gt;This is my first feed&lt;/b&gt;
```

c. Creating your feed

Let's create a basic feed. For example, let's say we have the following information we would like to announce.

Item 1:

101st AALL Annual Meeting and Conference

The 101st AALL Annual Meeting and Conference will be held July 12-15, 2008 in Portland Oregon, where the theme will be Energize, Explore, and Evolve!

<http://www.aallnet.org/events/>

Item 2:

2008 AALL Webmasters Workshop

The 2008 AALL Webmasters Workshop in Portland, Oregon, will be held on Saturday, July 12, 2008 at Kinetic Technology Solutions.

<http://www.aallnet.org/committee/aac/workshops/>

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rss version="2.0">
<channel>
<title>Basic RSS Feed</title>
<link>http://www.aallnet.org</link>
<description>This is a sample of a basic RSS feed for the 2008 AALL Webmasters Workshop.</
description>
  <item>
    <title>101st AALL Annual Meeting and Conference</title>
    <link>http://www.aallnet.org/events/</link>
    <description>The 101st AALL Annual Meeting and Conference will be held July 12-15, 2008 in
Portland Oregon, where the theme will be Energize, Explore, and Evolve!</description>
  </item>
  <item>
    <title>2008 AALL Webmasters Workshop</title>
    <link>http://www.aallnet.org/committee/aac/workshops/</link>
    <description>The 2008 AALL Webmasters Workshop in Portland, Oregon, will be held on
Saturday, July 12, 2008 at Kinetic Technology Solutions.</description>
  </item>
</channel>
</rss>
```

As you can see, an RSS feed is simply an XML file that contains certain required tags.

The first line, `<?xml version="1.0" encoding="ISO-8859-1"?>`, declares that you are using XML. Please note that nothing appears before this line – not even a whitespace. This is because certain browsers will return an error if there is anything before it.

Every RSS file has as its root the `<rss>` element. Following that, only one `<channel>` element is found. The channel is required to have a `<title>`, `<link>`, and `<description>`. The actual content of your feed appear as an `<item>`. Within each `<item>` you are required to have a `<title>`, `<link>`, and `<description>` as well. You will most likely end up with numerous `<item>` elements depending on how much content you are displaying.

d. Optional elements

There are several optional elements that may be included in the <channel> element and the <item> elements. Please refer to *RSS 2.0 Specification (Harvard Law)* at <http://cyber.law.harvard.edu/rss/rss.html> for more information on these optional elements.

I have included some helpful optional elements below in red.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<rss version="2.0">
<channel>
<title>Basic RSS Feed with Options</title>
<link>http://www.aallnet.org</link>
<description>This is a sample of a basic RSS feed, with optional elements, for the 2008 AALL
Webmasters Workshop.</description>
<language>en-us</language>
<copyright>Copyright 2008. American Association of Law Libraries. All Rights Reserved.</
copyright>
<lastBuildDate>Sat, 05 Jul 2008 09:00:00 CDT</lastBuildDate>
  <item>
    <title>101st AALL Annual Meeting and Conference</title>
    <link>http://www.aallnet.org/events/</link>
    <description>The 101st AALL Annual Meeting and Conference will be held July 12-15, 2008 in
Portland Oregon, where the theme will be Energize, Explore, and Evolve!</description>
    <pubDate>Fri, 04 Jul 2008 09:00:00 CDT</pubDate>
  </item>
  <item>
    <title>2008 AALL Webmasters Workshop</title>
    <link>http://www.aallnet.org/committee/aac/workshops/</link>
    <description>The 2008 AALL Webmasters Workshop in Portland, Oregon, will be held on
Saturday, July 12, 2008 at Kinetic Technology Solutions.</description>
    <pubDate>Fri, 04 Jul 2008 08:30:00 CDT</pubDate>
  </item>
</channel>
</rss>
```

The <language> element defines the language the feed is written in. This helps aggregators group feeds by language.

The <copyright> element provides the copyright content for the feed.

The <lastBuildDate> lets aggregators know when the content of the feed was last changed.

The <pubDate> indicates when an item was published.

*Please note that when providing date and times, they must be in the RSS specified format:

DayAbbreviation, DateofMonth MonthAbbreviation Year TimeInGMT

Example: Fri, 04 Jul 2008 09:00:00 CDT

DayAbbreviation:

The abbreviation for the day of the week.
 "Mon", "Tue", "Wed", "Thu", "Fri", "Sat", "Sun"

DateofMonth:

The number that represents the month of the year. It must be in a 2-digit format.
 Example: "01", "02"

MonthAbbreviation:

The abbreviation for the month of the year.
 "Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"

Year:

The number that represents the year. It must be in a 4-digit format.
 Example: "2008"

TimeInGMT:

This contains two parts - the time and the time zone.

- The first part is the time in Military time (00:00:00 - 23:59:59)
- The second part is the time zone
 - "EST", "EDT" - Eastern Time
 - "CST", "CDT" - Central Time
 - "MST", "MDT" - Mountain Time
 - "PST", "PDT" - Pacific Time

IV. Creating a Dynamic Feed

a. Using ASP to dynamically generating your feeds

We have just taken a look at manually creating a RSS feed. If your content changes once a day, or even a couple times a day, the static approach is probably your best option.

If your feed constantly changes throughout the day, changes without your knowledge, or comes from a database, a dynamically-generated feed may be the better option.

Using our earlier example, what if our information was found in a Microsoft Access database instead?

Table: Events

EventID	DateCreated	EventTitle	EventDetails	EventURL

1	7/4/2008 9:00:00 AM	101st AALL Annual Meeting and Conference	The 101st AALL Annual Meeting and Conference will be held July 12-15, 2008 in Portland Oregon, where the theme will be Energize, Explore, and Evolve!	http://www.aallnet.org/events/
2	7/4/2008 8:30:00 AM	2008 AALL Webmasters Workshop	The 2008 AALL Webmasters Workshop in Portland, Oregon, will be held on Saturday, July 12, 2008 at Kinetic Technology Solutions.	http://www.aallnet.org/committee/aac/workshops/

Let's make our feed dynamic. The first thing I did was copy and paste the "Basic RSS Feed with Options" into a blank ASP file. After that, I simply plugged in the necessary ASP code to pull the information from the database. I have highlighted the ASP portions in red.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<!-- #include file="rss-includes.asp" -->
<%
Response.Buffer = true
Response.ContentType = "text/xml"
%>
<rss version="2.0">
<channel>
<title>RSS Feed with a database</title>
<link>http://www.aallnet.org/</link>
<description>This is a sample RSS feed that automatically updates itself by pulling information
from a Microsoft Access database.</description>
<language>en-us</language>
<copyright>Copyright <%=Year(Now())%> American Association of Law Libraries. All Rights
Reserved.</copyright>
<lastBuildDate><%=FormatRSSDate((Now()))%></lastBuildDate>

<%
'delcare variables
Dim dbConnection
Dim rsConnection

'create ADO Connection object
Set dbConnection = Server.CreateObject("ADODB.Connection")

'establish communication with database
dbConnection.Open "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" & Server.MapPath
("folderName/MySampleDatabase.mdb")

'execute SQL query
Set rsConnection = dbConnection.Execute("SELECT DateCreated, EventTitle, EventDetails,
EventURL FROM Events ORDER BY DateCreated DESC")

intCount = 0

'create <item>'s for feed and display the most recent 10 events

Do While intCount < 10
%>

```

```

<item>
<title><%=ApplyXMLFormatting(rsConnection("EventTitle"))%></title>
<link><%=ApplyXMLFormatting(rsConnection("EventURL"))%></link>
<description><%=ApplyXMLFormatting(rsConnection("EventDetails"))%></description>
<pubDate><%=FormatRSSDate(rsConnection("DateCreated"))%></pubDate>
</item>
<%
intCount = intCount + 1
rsConnection.MoveNext
Loop

**close recordset connection**
rsConnection.Close
Set rsConnection = Nothing

**close database connection**
dbConnection.close
Set dbConnection = Nothing
%>

</channel>
</rss>

```

As we learned earlier, the first line is `<?xml version="1.0" encoding="ISO-8859-1"?>`; there is nothing that appears before this line.

The second line refers to an include file. This line is completely optional. I created some functions and kept them separate from the RSS feed. This way, if I have multiple feeds, they can all point to the one "include" file. If I make a change to that include file, it will automatically update "across the board." If I have kept the functions within the RSS feed, I would have to update each file. We'll visit the functions in moment.

```

<!-- #include file="rss-includes.asp" -->

```

In the next couple of lines, we declare some ASP settings. We make sure the Buffer is set to "true" because we want the server to completely process the ASP script before it sends any response back. If the Buffer was "false," then you would receive an error message. Next, we want to declare that we are using XML, so we set the ContentType to "text/xml." In general, the ContentType is set to "text/html."

```

<%
Response.Buffer = true
Response.ContentType = "text/xml"
%>

```

In the next set of lines, we declare our RSS elements. The only noticeable change is to the `<copyright>` and `<lastBuildDate>` elements. The current year is now automatically placed in the `<copyright>` using the standard VBScript Date/Time Functions "Year()" and "Now()". Since this RSS feed is dynamically generated, we set the `<lastBuildDate>` to whatever time the feed is accessed.

```
<copyright>Copyright <%=Year(Now())%> American Association of Law Libraries. All Rights Reserved.</copyright>
```

```
<lastBuildDate><%=FormatRSSDate((Now()))%></lastBuildDate>
```

Next, we will work on the content of the RSS feed. The first thing we do is establish a connection to our database and access the information we need from our table.

```
'declare variables
Dim dbConnection
Dim rsConnection

'create ADO Connection object
Set dbConnection = Server.CreateObject("ADODB.Connection")

'establish communication with database
dbConnection.Open "Provider=Microsoft.Jet.OLEDB.4.0; Data Source=" & Server.MapPath
("folderName/MySampleDatabase.mdb")

'execute SQL query
Set rsConnection = dbConnection.Execute("SELECT DateCreated, EventTitle, EventDetails,
EventURL FROM Events ORDER BY DateCreated DESC")
```

Once we pull our information from the table, we simply plug the information into each <item> element. You will notice that the <item> elements are placed in a standard Do While Loop. This Loop is set up to retrieve the 10 most recent items from the table. If there are not even 10 items in the table, or after 10 items have been placed, it will automatically exit the Loop.

```
intCount = 0

'create <item>'s for feed and display the most recent 10 events

Do While intCount < 10
  %>
  <item>
  <title><%=ApplyXMLFormatting(rsConnection("EventTitle"))%></title>
  <link><%=ApplyXMLFormatting(rsConnection("EventURL"))%></link>
  <description><%=ApplyXMLFormatting(rsConnection("EventDetails"))%></description>
  <pubDate><%=FormatRSSDate(rsConnection("DateCreated"))%></pubDate>
  </item>
  <%
  intCount = intCount + 1
  rsConnection.MoveNext
Loop

'**close recordset connection**
rsConnection.Close
Set rsConnection = Nothing

'**close database connection**
dbConnection.close
Set dbConnection = Nothing
%>
```

```
</channel>  
</rss>
```

b. RSS functions used

i. ApplyXMLFormatting

As we learned earlier, we cannot use certain characters in XML files. Because these special characters may exist in the EventTitle, EventDetails, and EventURL fields, the values are passed through the ApplyXMLFormatting() Function. The function will automatically replace the forbidden characters with their acceptable equivalents.

```
Function ApplyXMLFormatting(strInput)
```

```
    strInput = Replace(strInput,"&", "&amp;")  
    strInput = Replace(strInput,"'", "&apos;")  
    strInput = Replace(strInput,"\"", "&quot;")  
    strInput = Replace(strInput, ">", "&gt;")  
    strInput = Replace(strInput,"<", "&lt;")
```

```
    ApplyXMLFormatting = strInput
```

```
End Function
```

ii. FormatRSSDate

As we learned earlier, date and time values must be in the RSS specified format. The FormatRSSDate() Function will automatically convert all date and time values passed through into the proper format.

```
Function FormatRSSDate(byVal strDate)
```

```
    'format date received in  
    'e.g. 1/1/2008 8:00:00 AM
```

```
    Dim tmpWeekdayName  
    Dim tmpDay  
    Dim tmpMonth  
    Dim tmpYear  
    Dim tmpHour  
    Dim tmpMinute  
    Dim tmpSecond  
    Dim tmpFinal
```

```
    'get the abbreviated weekday name (e.g. sat, sun, mon)  
    tmpWeekdayName = WeekDayName(DatePart("w",strDate,1),True)
```

```
    'get the day (e.g. 1, 2, 3)  
    tmpDay = DatePart("d",strDate)  
    If Len(tmpDay) < 2 then tmpDay = "0" & tmpDay
```

```

'get the abbreviated month name (e.g. jan, feb, mar)
tmpMonth = MonthName(DatePart("m", strDate), True)

'get the year
tmpYear = DatePart("yyyy",strDate)

'hour in 24hr format
tmpHour = DatePart("h",strDate)
If Len(tmpHour) < 2 then tmpHour = "0" & tmpHour

'minutes
tmpMinute = DatePart("n",strDate)
If Len(tmpMinute) < 2 then tmpMinute = "0" & tmpMinute

'seconds
tmpSecond = DatePart("s",strDate)
If Len(tmpSecond) < 2 then tmpSecond = "0" & tmpSecond

'concatenate everything together (e.g. Sat, 07 Sep 2008 00:00:01 CDT)
tmpFinal = tmpWeekdayName & ", " & tmpDay & " " & tmpMonth & " " & tmpYear   tmpFinal =
tmpFinal & " " & tmpHour & ":" & tmpMinute & ":" & tmpSecond & " CDT"

fncFmtDate = tmpFinal

End Function

```

V. Marketing Your Feed

Once you have created your feed, it is important to let your audience know that your feed exists. Simply burying the feed in your Web site will not help you. Web sites will include a URL on their Web site that will direct viewers to a Web page that will list all of their feeds. Typically, they will include a standard RSS image with the URL to make it stand out.



For example:  [Add our RSS Feeds](#)