

XML Workshop XVIII - Generating an RSS 2.0 Feed with TSQL

By [Jacob Sebastian](#), 2008/03/26

Introduction

In the previous sessions of [XML Workshop](#), we have seen several XML processing examples. Some of the examples demonstrated how to generate XML output in a specific structure. In this session of XML Workshop, let us see how to generate an RSS feed using TSQL. Most web sites today adhere to the Web 2.0 standards and RSS/ATOM feeds are an integral part of them. If you are not familiar with RSS feeds, you can find a basic introduction [here](#). RSS has several versions and the one that is widely used today is version 2.0. You can find the documentation of RSS 2.0 [here](#).

Most of the times, web applications generate feeds at the application layer. The feed generator would execute a query/stored-procedure, fetch the data and generate the XML document using custom application code or a third party library like [RSS.NET](#). The purpose of this session is to look deeper into the XML capabilities of SQL Server 2005 and see if it could generate an RSS 2.0 with a FOR XML operation. The examples and code given in this article is purely for the purpose of explaining how XML data can be generated in TSQL.

As mentioned earlier, an RSS 2.0 feed should follow a certain structure and rules. There are certain mandatory elements and attributes. Certain values like pubDate should be in a specific format. Most applications that read RSS feeds (RSS Readers) validate the feed against the rules given in the [RSS specification](#)

and will reject the feed if it does not follow the rules defined in the specification. An online feed validator like [FeedValidator.org](#)

can be used to validate the RSS feed to make sure that it follows all the rules defined by the [RSS specification](#). In this session, we will generate an RSS 2.0 feed and validate it with [FeedValidator.org](#).

Sample Feed

For the purpose of this example, let us assume that we need to create an RSS 2.0 feed that contains all the articles in the [XML Workshop](#) series. To keep the examples simple, we will process only two records. Here is the output that we expect to generate by the end of this LAB.

```

<?xml version="1.0" encoding="utf-8" ?>
<rss version="2.0">
  <channel>
    <title>Welcome to XML Workshop</title>

    <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
    <description>A collection of short articles on SQL Server and XML</description>
    <webMaster>jacob@dotnetquest.com (Jacob Sebastian)</webMaster>
    <language>en-us</language>
    <image>
      <url>http://www.sqlserverandxml.com/image.jpg</url>
      <title>Welcome to XML Workshop</title>

    <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
    <width>144</width>
    <height>22</height>
  </image>
  <copyright>Jacob Sebastian. All rights reserved.</copyright>

```

```

<lastBuildDate>Wed, 12 Mar 2008 23:45:02 GMT</lastBuildDate>
<ttl>100</ttl>
<item>
  <title>XML Workshop I - Generating XML with FOR XML</title>
  <link>http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</link>
  <description>A short article that explains how to generate XML output with TSQL
keyword FOR XML</description>
  <guid
isPermaLink="true">http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</gui
  <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
</item>
<item>
  <title>XML Workshop II - Reading values from XML variables</title>
  <link>http://www.sqlservercentral.com/articles/Miscellaneous/2996/</link>
  <description>This article explains how to read values from an XML variable using
XQuery</description>
  <guid
isPermaLink="true">http://www.sqlservercentral.com/articles/Miscellaneous/2996/</guid>
  <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
</item>
</channel>
</rss>

```

Before we proceed further, we need to make sure that the XML output that we intend to generate is a valid RSS 2.0 document. You could test this by using feedvalidator.org. Open a browser window and navigate to feedvalidator.org. Enter the [url](#) of the above sample RSS 2.0 feed and click on the "validate" button.

Sample Tables and Data

Let us create two tables to store the data needed for this LAB. We need one table to store the information about the RSS Channel and another table for storing the data of each RSS item. Here is the script for those tables.

```

CREATE TABLE channel(
  Title VARCHAR(100),
  Link VARCHAR(100),
  Description VARCHAR(200),
  WebMaster VARCHAR(50),
  Language VARCHAR(20),
  ImageUrl VARCHAR(100),
  ImageTitle VARCHAR(100),
  ImageLink VARCHAR(100),
  ImageWidth SMALLINT,
  ImageHeight SMALLINT,
  CopyRight VARCHAR(100),
  LastBuildDate DATETIME,
  ttl SMALLINT )
GO

CREATE TABLE Articles(
  Title VARCHAR(100),
  Link VARCHAR(100),
  Description VARCHAR(200),
  Guid VARCHAR(100),
  PubDate DATETIME )
GO

```

Here is the code to populate the tables with some sample data

```

INSERT INTO channel (
  Title,
  Link,
  Description,
  Webmaster,
  Language,
  ImageUrl,

```

```

        ImageTitle,
        ImageLink,
        ImageWidth,
        ImageHeight,
        CopyRight,
        LastBuildDate,
        ttl)
SELECT
    'Welcome to XML Workshop',
    'http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html',
    'A collection of short articles on SQL Server and XML',
    'jacob@dotnetquest.com (Jacob Sebastian)',
    'en-us',
    'http://www.sqlserverandxml.com/image.jpg',
    'Welcome to XML Workshop',
    'http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html',
    144,
    22,
    'Jacob Sebastian. All rights reserved.',
    '2008-03-12 23:45:02',
    100

INSERT INTO Articles (
    Title,
    Link,
    Description,
    Guid,
    PubDate )
SELECT
    'XML Workshop I - Generating XML with FOR XML',
    'http://www.sqlservercentral.com/columnists/jSebastian/2982.asp',
    'A short article that explains how to generate XML output with TSQL keyword FOR
XML',
    'http://www.sqlservercentral.com/columnists/jSebastian/2982.asp',
    '2008-03-12 23:45:02'
UNION ALL
SELECT
    'XML Workshop II - Reading values from XML variables',
    'http://www.sqlservercentral.com/articles/Miscellaneous/2996/',
    'This article explains how to read values from an XML variable using XQuery',
    'http://www.sqlservercentral.com/articles/Miscellaneous/2996/',
    '2008-03-12 23:45:02'

```

Generating the feed

Let us start with the item element, which is pretty much easy. Here is the code that generates the item elements.

```

SELECT
    Title AS title,
    Link AS link,
    Description AS description,
    'true' AS 'guid/@isPermaLink',
    Guid AS guid,
    PubDate AS pubDate
FROM Articles FOR XML PATH('item'), TYPE

```

The above code produces the following XML document.

```

<item>
  <title>XML Workshop I - Generating XML with FOR XML</title>
  <link>http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</link>
  <description>A short article that explains how to generate XML output with TSQL
keyword FOR XML</description>

```

```

    <guid
isPermaLink="true">http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</gui
    <pubDate>2008-03-12T23:45:02</pubDate>
</item>
<item>
    <title>XML Workshop II - Reading values from XML variables</title>
    <link>http://www.sqlservercentral.com/articles/Miscellaneous/2996/</link>
    <description>This article explains how to read values from an XML variable using
XQuery</description>
    <guid
isPermaLink="true">http://www.sqlservercentral.com/articles/Miscellaneous/2996/</guid>
    <pubDate>2008-03-12T23:45:02</pubDate>
</item>

```

Though the XML looks good, there is a problem. The format of the date value (pubDate) is not correct. RSS 2.0 requires that the date value should be in [RFC 822 date format](#). So we need to format the date value to a valid RFC 822 date value. Here is the modified version of the code.

```

SELECT
    Title AS title,
    Link AS link,
    Description AS description,
    'true' AS 'guid/@isPermaLink',
    Guid AS guid,
    LEFT(DATENAME(dw, PubDate),3) + ', ' +
        STUFF(CONVERT(nvarchar, PubDate, 113), 21, 4, ' GMT')
        AS pubDate
FROM Articles FOR XML PATH('item'), TYPE

```

Here is the corrected XML.

```

<item>
    <title>XML Workshop I - Generating XML with FOR XML</title>
    <link>http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</link>
    <description>A short article that explains how to generate XML output with TSQL
keyword FOR XML</description>
    <guid
isPermaLink="true">http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</gui
    <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
</item>
<item>
    <title>XML Workshop II - Reading values from XML variables</title>
    <link>http://www.sqlservercentral.com/articles/Miscellaneous/2996/</link>
    <description>This article explains how to read values from an XML variable using
XQuery</description>
    <guid
isPermaLink="true">http://www.sqlservercentral.com/articles/Miscellaneous/2996/</guid>
    <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
</item>

```

Now, let us write the query to generate the "channel" element.

```

SELECT
    Title AS title,
    Link AS link,
    Description AS description,
    Webmaster AS webMaster,
    Language AS language,
    ImageUrl AS 'image/url',
    ImageTitle AS 'image/title',
    ImageLink AS 'image/link',
    ImageWidth AS 'image/width',
    ImageHeight AS 'image/height',
    CopyRight AS copyright,
    LEFT(DATENAME(dw, LastBuildDate),3) + ', ' +
        STUFF(CONVERT(nvarchar, LastBuildDate, 113), 21, 4, ' GMT')
        AS lastBuildDate,

```

```

Ttl AS ttl,
(
    SELECT
        Title AS title,
        Link AS link,
        Description AS description,
        'true' AS 'guid/@isPermaLink',
        Guid AS guid,
        LEFT(DATENAME(dw, PubDate),3) + ', ' +
            STUFF(CONVERT(nvarchar, PubDate, 113), 21, 4, ' GMT')
            AS pubDate
    FROM Articles FOR XML PATH('item'), TYPE
)
FROM channel
FOR XML PATH('channel'), TYPE

```

```

<channel>
  <title>Welcome to XML Workshop</title>

  <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
  <description>A collection of short articles on SQL Server and XML</description>
  <webMaster>jacob@dotnetquest.com (Jacob Sebastian)</webMaster>
  <language>en-us</language>
  <image>
    <url>http://www.sqlserverandxml.com/image.jpg</url>
    <title>Welcome to XML Workshop</title>

  <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
    <width>144</width>
    <height>22</height>
  </image>
  <copyright>Jacob Sebastian. All rights reserved.</copyright>
  <lastBuildDate>Wed, 12 Mar 2008 23:45:02 GMT</lastBuildDate>
  <ttl>100</ttl>
  <item>
    <title>XML Workshop I - Generating XML with FOR XML</title>
    <link>http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</link>
    <description>A short article that explains how to generate XML output with TSQ
keyword FOR XML</description>
    <guid
isPermaLink="true">http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</gui
    <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
  </item>
  <item>
    <title>XML Workshop II - Reading values from XML variables</title>
    <link>http://www.sqlservercentral.com/articles/Miscellaneous/2996/</link>
    <description>This article explains how to read values from an XML variable usi
XQuery</description>
    <guid
isPermaLink="true">http://www.sqlservercentral.com/articles/Miscellaneous/2996/</guid>
    <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
  </item>
</channel>

```

The next step is to add the root element and generate the xml header. here is the final version of the code.

```

SELECT
'<?xml version="1.0" encoding="utf-8" ?>' +
( SELECT
  '2.0' AS '@version',
  (
    SELECT
      Title AS title,
      Link AS link,
      Description AS description,
      Webmaster AS webMaster,
      Language AS language,
      ImageUrl AS 'image/url',
      ImageTitle AS 'image/title',

```

```

ImageLink AS 'image/link',
ImageWidth AS 'image/width',
ImageHeight AS 'image/height',
CopyRight AS copyright,
LEFT(DATENAME(dw, LastBuildDate),3) + ', ' +
    STUFF(CONVERT(nvarchar,LastBuildDate,113),21,4,' GMT')
    AS lastBuildDate,
Ttl AS ttl,
(
    SELECT
        Title AS title,
        Link AS link,
        Description AS description,
        'true' AS 'guid/@isPermaLink',
        Guid AS guid,
        LEFT(DATENAME(dw, PubDate),3) + ', ' +
            STUFF(CONVERT(nvarchar,PubDate,113),21,4,' GMT')
            AS pubDate
        FROM Articles FOR XML PATH('item'), TYPE
    )
FROM channel
FOR XML PATH('channel'), TYPE
)
FOR XML PATH('rss') )

```

```

<?xml version="1.0" encoding="utf-8" ?>
<rss version="2.0">
  <channel>
    <title>Welcome to XML Workshop</title>

    <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
    <description>A collection of short articles on SQL Server and XML</description>
    <webMaster>jacob@dotnetquest.com (Jacob Sebastian)</webMaster>
    <language>en-us</language>
    <image>
      <url>http://www.sqlserverandxml.com/image.jpg</url>
      <title>Welcome to XML Workshop</title>

    <link>http://www.sqlserverandxml.com/2007/12/xml-workshop-at-sql-server-central.html</
      <width>144</width>
      <height>22</height>
    </image>
    <copyright>Jacob Sebastian. All rights reserved.</copyright>
    <lastBuildDate>Wed, 12 Mar 2008 23:45:02 GMT</lastBuildDate>
    <ttl>100</ttl>
    <item>
      <title>XML Workshop I - Generating XML with FOR XML</title>
      <link>http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</link>
      <description>A short article that explains how to generate XML output with
keyword FOR XML</description>
      <guid
isPermaLink="true">http://www.sqlservercentral.com/columnists/jSebastian/2982.asp</gui
      <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
    </item>
    <item>
      <title>XML Workshop II - Reading values from XML variables</title>
      <link>http://www.sqlservercentral.com/articles/Miscellaneous/2996/</link>
      <description>This article explains how to read values from an XML variable
using XQuery</description>
      <guid
isPermaLink="true">http://www.sqlservercentral.com/articles/Miscellaneous/2996/</guid>
      <pubDate>Wed, 12 Mar 2008 23:45:02 GMT</pubDate>
    </item>
  </channel>
</rss>

```

Conclusions

This is yet another session that demonstrates an XML shaping example. We have seen different XML shaping requirements and their implementation in the previous sessions of the [XML Workshop](#). This session explains the basics of generating an RSS 2.0 feed using TSQL keyword FOR XML PATH.

Copyright © 2002-2008 Simple Talk Publishing. All Rights Reserved. [Privacy Policy](#). [Terms of Use](#)