

## OPAC: New Books RSS Feed Embedded in Search Page

Created By: Jim Robinson  
Created on: 5/14/2019

[This article was copied from the Voyager Wiki.]

### Description

A perl script generates an RSS feed for a limited number of new books from Michael Doran's New Books program and embedded in a WebVoyage 7 search page.

Additional Author: Michael Doran



Download at [https://library.tccd.edu/code/webvoyage/new\\_books\\_subjects.php](https://library.tccd.edu/code/webvoyage/new_books_subjects.php)

A [Full RSS Feeder](#) is also available. Just remember to set up the configuration at the top of the script.

The [RSS Feed](#) can be seen in the lower right corner of our Basic Search page.

Our [Full RSS Feed is also available](#). It contains 1 week of new books sorted by title.

# Installation Instructions

Michael Doran's New Books program must be installed and working to use this script.

1. Edit new\_books\_mini.cgi between **Configuration** and **End Configuration**. Required items include:

- Your library domain
- The number of books you want to appear in your RSS feed.
- The number of books desired from each location
- The number of weeks to go back looking for books (usually between 1 and 4)
- The name of your RSS feed
- A brief description of your RSS feed
- The names of desired locations as found in newBooks.txt
- If desired, image information for your RSS feed

2. Upload new\_books\_mini.cgi to /m1/voyager/**xxxdb**/tomcat/vwebv/context/vwebv/htdocs/vwebv/new\_books\_mini.cgi, which is where your newBooks.cgi script should be located.

3. Set the permissions to 755 (rwxr-xr-x)

4. Open Firefox and test the [http://your.library.domain/vwebv/new\\_books\\_mini.cgi](http://your.library.domain/vwebv/new_books_mini.cgi) (be sure to change **your.library.domain** to your library domain)

5. Back up /m1/voyager/**xxxdb**/tomcat/vwebv/context/vwebv/ui/**skin**/xsl/pageTools/tools.xsl

6. In tools.xsl, find

```
<xsl:variable name="uppercase" select="'ABCDEFGHIJKLMNOPQRSTUVWXYZ' " />
```

and add the following lines underneath it (be sure to change **your.library.domain** to your library domain):

```
<xsl:variable name="uppercase" select="'ABCDEFGHIJKLMNOPQRSTUVWXYZ' " />
<xsl:variable name="NewBooks" select="document ('http://your.library.domain/vwebv/new_books_
mini.cgi')"/>
<xsl:variable name="new_books" select="$NewBooks/rss/channel"/>
```

7. In tools.xsl, find

```
<xsl:if test="$Configs/pageConfigs/pageHTML/page \ [@name=$pageNameId] \ [@position=$position] ">
  <div class="pageHTMLSnippet">
    <xsl:copy-of select="$Configs/pageConfigs/pageHTML/
page \ [@name=$pageNameId] \ [@position=$position] /node () " />
  </div>
```

```
</xsl:if>
```

and change it to look like this (without the line numbers, of course):

```
01 <xsl:if test="$Configs/pageConfigs/pageHTML/  
page\[@name=$pageNameId\]\[@position=$position\]">  
02   <div class="pageHTMLSnippet">  
03     <xsl:copy-of select="$Configs/pageConfigs/pageHTML/  
page\[@name=$pageNameId\]\[@position=$position\]/node()" />  
04   </div>  
05  
06   <xsl:if test="$pageNameId='page.searchBasic'">  
07     <div id="new_books">  
08       <ul>  
09         <xsl:for-each select="$new_books/item">  
10           <li>  
11             <xsl:attribute name="class"><xsl:value-of select="description"/></  
xsl:attribute>  
12             <a>  
13               <xsl:attribute name="href"><xsl:value-of select="link"/></xsl:attribute>  
14               <xsl:attribute name="title"><xsl:value-of select="description"/></  
xsl:attribute>  
15               <xsl:value-of select="title"/>  
16             </a>  
17           </li>  
18         </xsl:for-each>  
19       </ul>  
20     </div>  
21   </xsl:if>  
22 </xsl:if>
```

8. Edit the appropriate stylesheets (e.g., searchBasic.css). See tips below for stylesheet hints.

- Line 6 and line 21 are used to display the feed only on the Basic Search page.
- The class is the location (which is the description). If you have multiple words in your locations, then you have multiple classes with which to work. Choose a common class name to display common features and unique class names to differentiate. For example, our locations are "Northeast Library Circulating Collection", "Northwest Library Circulating Collection", etc. In our stylesheet, .Northeast and .Northwest have different colors, while .Circulating has a common padding.

- Because our feed only appears in the Basic Search, my CSS markup only appears in searchBasic.css. I have a separate CSS page for newBooks.cgi in which I have
 

```
#new_books { display:none;}
```

 so that the feed doesn't appear on the newBooks.cgi page, which uses the Basic Search page as a "template".
- Remember to change items in ***bold italics*** to your institution information

- Download and install [Firefox](#) with [Firebug](#)

TO DO list

None

Known issues

None

Comments

None

Added by [Jim Robinson](#) on Apr 03, 2009 20:19, last edited by [Jim Robinson](#) on May 11, 2009 17:06([Detailed page information](#))



```
<xsl:variable name="uppercase" select="ABCDEFGHIJKLMNOPQRSTUVWXYZ" />
```

and add the following lines underneath it (be sure to change *your.library.domain* to your library domain):

```
<xsl:variable name="uppercase" select="ABCDEFGHIJKLMNOPQRSTUVWXYZ" />  
<xsl:variable name="NewBooks" select="document('http://your.library.domain/web/new_books_min1.cgi')"/>  
<xsl:variable name="new_books" select="$NewBooks/rss/channel"/>
```

In tools.xml, find

```
<xsl:if test="$Configs/pageConfigs/pageHTML/page\[@name=$pageNameId\]\[@position=$position\]">  
  <div class="pageHTMLSnippet">  
    <xsl:copy-of select="$Configs/pageConfigs/pageHTML/page\[@name=$pageNameId\]\[@position=$position\]/node()"/>  
  </div>  
</xsl:if>
```

and change it to look like this (without the line numbers, of course):

```
81 <xsl:if test="$Configs/pageConfigs/pageHTML/page\[@name=$pageNameId\]\[@position=$position\]">  
82   <div class="pageHTMLSnippet">  
83     <xsl:copy-of select="$Configs/pageConfigs/pageHTML/page\[@name=$pageNameId\]\[@position=$position\]/node()"/>  
84   </div>  
85   <xsl:if test="$pageNameId='page.searchBasic'">  
86     <div id="new_books">  
87       <ul>  
88         <xsl:for-each select="$new_books/item">  
89           <li>  
90             <xsl:attribute name="class"><xsl:value-of select="description"/></xsl:attribute>  
91             <a>  
92               <xsl:attribute name="href"><xsl:value-of select="link"/></xsl:attribute>  
93               <xsl:attribute name="title"><xsl:value-of select="description"/></xsl:attribute>  
94               <xsl:value-of select="title"/>  
95             </a>  
96           </li>  
97         </xsl:for-each>  
98       </ul>  
99     </div>  
100   </xsl:if>  
101 </xsl:if>
```

## [Report](#)