

Configuring Letters in Alma

Alan Dawson – Bangor University / WHELF

a.f.dawson@bangor.ac.uk



What I am going to cover

Tools I use

Relationships

How to view
the email XML

Library Account
Statement
Letter

Sender
Receiver

How to stop
emails going to
expired users

Choosing
Language

Questions and
Answers

Tools I use

Alma

Notepad++

Internet Explorer

Google

Comments

HTML / CSS


Relationships

Not all data is equal

We don't have everything in one Letter

The Letters are made up of code from elsewhere

Header

 **Gwasanaeth Llyfrgell
Library Service**

Library Account Statement 15/06/2021

21000023832236

Dawson, Alan

Information Serv Library

Bangor University

College Road

LL57 2DG

Bangor University
Bangor University
Main Library
Bangor
LL57 2DG

Sender
Receiver

Mail Reason

Dear MR Alan Dawson

Your account is due to close in the next 16 days. Please ensure you return your books and pay your fines before you leave. If you have any questions please contact library@bangor.ac.uk for more information.

Below is a summary of your library account, including items you are currently borrowing and any active fines.

Library Fine: £55.50 GBP

Please note: If your fines exceed £10 you will not be able to borrow or reserve items and your loans will stop automatically renewing which could result in further charges. If you have any concerns, please contact the library.

Overdue Loans - Please contact the library

Borrowing
Activity Email

Footer

How to view the email XML

```
for object to mirror_...
mirror_mod.mirror_object

operation == "MIRROR_X":
mirror_mod.use_x = True
mirror_mod.use_y = False
mirror_mod.use_z = False
operation == "MIRROR_Y":
mirror_mod.use_x = False
mirror_mod.use_y = True
mirror_mod.use_z = False
operation == "MIRROR_Z":
mirror_mod.use_x = False
mirror_mod.use_y = False
mirror_mod.use_z = True

#selection at the end -add
mirror_ob.select= 1
mirror_ob.select=1
context.scene.objects[one.name].select= 1
mirror_ob.select = 0
= bpy.context.selected_objects[one.name]
data.objects[one.name].select= 1

print("please select exactly one object")

--- OPERATOR CLASSES ---

types.Operator):
X mirror to the selected
object.mirror_mirror_x"
mirror X"
```

Library Account Statement

Library Debt

Items needing action

Current Loans



Some code

Mail Reason

Applies across all our
emails

Needed to be updated to
provide preferred name

Expiry Warning



Some code

How to stop emails going to expired users

This is in Bangor's footer

Useful as we found that running the borrowing activity job sent emails to expired users

Alas we don't have personal email addresses set for our users

Drop in Code for stopping emails to expired users

```
<!--Perform check to confirm email can be sent - this checks to see if a user has already expired this is required due to the sending of the borrowing activity report-->
<table cellpadding="5" cellspacing="0" border="0">
<xsl:if test="notification_data/general_data/current_date">
<xsl:param name="currentDateEC" select="notification_data/general_data/current_date"/>
<xsl:param name="expiryDateEC" select="notification_data/receivers/receiver/user/expiry_date"/>
<!--Split the current date-->
<xsl:variable name="CDAYEC" select="substring($currentDateEC,1,2)"/>
<xsl:variable name="CMONEC" select="substring($currentDateEC,4,2)"/>
<xsl:variable name="CYEAEC" select="substring($currentDateEC,7,4)"/>
<!--Split the expiry date-->
<xsl:variable name="EDAYEC" select="substring($expiryDateEC,1,2)"/>
<xsl:variable name="EMONEC" select="substring($expiryDateEC,4,2)"/>
<xsl:variable name="EYEAEC" select="substring($expiryDateEC,7,4)"/>
<!--Create dates for checking-->
<xsl:variable name="currentDate" select="concat($CYEAEC,$CMONEC,$CDAYEC)"/>
<xsl:variable name="expireddate" select="concat($EYEAEC,$EMONEC,$EDAYEC)"/>
<xsl:if test="$currentDate &gt; $expireddate">
<h4>User Expired - do not send email</h4>
<xsl:message terminate="yes">user has expired </xsl:message>
</xsl:if>
</xsl:if>
</table>
</xsl:template>
<!--Finish expired user check-->
```

Targeting a user group

- You need to know your user group IDs (we use Numbers at Bangor you can view them in Alma)
- If you start with a `<xsl:if>` you need to end at the other end with one `</xsl:if>`
- `<!-- BU Library Staff-->`
- `<xsl:if`
`test="(//notification_data/receivers/receiver/user/user_group = '49') and ($daystoexpiryC`
`<= '45')">`
`<p>We're sorry your leaving us in the next`
`<xsl:value-of select="$daystoexpiry"/> if Alma`
`has got this wrong let me know</p>`
`</xsl:if>`

Choosing Language

- This is fairly easy to do
- Just need to know the language codes

```
<!-- English -->
<xsl:if
test="/notification_data/receivers/receiver/pr
eferred_language = 'en'">
<p>Text</p>
</xsl:if>
<!-- English -->
<xsl:if
test="/notification_data/receivers/receiver/pr
eferred_language != 'en'">
<p></p>
```



Demonstrate

- Add extra columns to table
- How to test changes in Alma (sandbox)

The background is a blurred image of a library or study area with bookshelves. In the foreground, a stack of several books is visible, with the top one open. Floating around the books and text are various white, hand-drawn style symbols including mathematical signs like plus (+), minus (-), multiplication (x), and division (/), as well as letters like 'y' and 'x', and question marks. Some symbols are larger and more prominent than others.

Questions and Answers

Alan Dawson – Bangor University / WHELF

a.f.dawson@bangor.ac.uk

Borrowing Activity email

```
<?xml version="1.0" encoding="utf-8"?>
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:include href="header.xsl" />
<xsl:include href="senderReceiver.xsl" />
<xsl:include href="mailReason.xsl" />
<xsl:include href="footer.xsl" />
<xsl:include href="style.xsl" />
<xsl:include href="recordTitle.xsl" />

<xsl:template match="/">
<html>
<xsl:if test="notification_data/languages/string">
<xsl:attribute name="lang">
<xsl:value-of select="notification_data/languages/string"/>
</xsl:attribute>
</xsl:if>

<head>
<title>
<xsl:value-of select="notification_data/general_data/subject"/>
</title>

<xsl:call-template name="generalStyle" />
```

Mail reason

```
<?xml version="1.0" encoding="utf-8"?>
<!--Last updated 17-03-2021 - AFD to work around preferred names and also added title functionality-->
<xsl:stylesheet version="1.0"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform">

<xsl:template name="toWhomIsConcerned">
<table cellpadding="5" cellspacing="0" border="0">
<style>
.standardFont{
text-decoration: none;
font-size: 1em;
font-family: arial;
}
.UserAlert a{
color:white !important;
font-family: arial;
}
</style>
<tr>
<td>

<xsl:for-each select="notification_data">
<div class="standardFont">
<!--Check for Preferred First Name-->
<xsl:choose>
```