

HowToFormsUsingExternalHandler

Why?

- Cforms are hard to do. In fact, I haven't been able to get them going in any useful way. Until such time as [ProposalPublets](#) has been done, this is a nice easy weay to get form-to-email up.
- You don't need to do much customisation from the default publication to make this work.
- This lets Lenya do CMS which it is good at, and lets something else handle the email sending part.
- This is much easier for a novice lenya user to acheive.
- users can still edit the text in the page above the form using BXE or whatever.

How?

We will call the new form "contact".

In your publication's paramater-doctype.xmap, do this.

```
<map:components>
  <map:generators default="file"/>
  <map:transformers default="xslt"/>
  <map:readers default="resource"/>
  <map:serializers default="html"/>
  <map:matchers default="wildcard"/>
  <map:actions>
    <map:action logger="sitemap.action.sourcetype" name="sourcetype" src="org.apache.cocoon.acting.sourcetype.
SourceTypeAction">
      <sourcetype name="xhtml">
        <document-element namespace="http://www.w3.org/1999/xhtml"/>
      </sourcetype>
      <sourcetype name="links">
        <document-element namespace="http://apache.org/lenya/pubs/default/1.0"/>
      </sourcetype>
+      <sourcetype name="contact">
+        <document-element namespace="http://www.my_organisation.com/lenya/contactform"/>
+      </sourcetype>
    </map:action>
  </map:actions>
</map:components>
```

In your publication's xslt folder, add a new file contact2xhtml.xsl with the following contents:

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

<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
>

<xsl:import href="xhtml2xhtml.xsl"/>

</xsl:stylesheet>
```

In your publication's xslt folder, add a new file page2xhtml-content.xsl. Copy and paste one of your other templates and edit it. Note that in the example below, I use a file called common.xsl and centrally manage lots of my interface in there, in one place. This saves headaches.

```

<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE xsl:stylesheet [
<!ENTITY nbsp '&#160;'>
]>

<xsl:stylesheet version="1.0"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns="http://www.w3.org/1999/xhtml"
  xmlns:xhtml="http://www.w3.org/1999/xhtml"
  xmlns:page="http://apache.org/cocoon/lenya/cms-page/1.0"
  xmlns:lenya="http://apache.org/cocoon/lenya/page-envelope/1.0"
  xmlns:dc="http://purl.org/dc/elements/1.1/"
  xmlns:dcterms="http://purl.org/dc/terms/"
  xmlns:il8n="http://apache.org/cocoon/il8n/2.1"
  exclude-result-prefixes="page xhtml"
>

<xsl:param name="root"/>
<xsl:param name="document-id"/>
<xsl:param name="raw"/>
<xsl:param name="document-type"/>
<xsl:param name="document-label"/>
<xsl:param name="url"/>
<xsl:param name="language"/>
<xsl:include href="common.xsl"/>

<xsl:template match="cmsbody">
<html>
  <head>
    <title><xsl:value-of select="$document-label"/></title>
    <link href="{ $root }/css/styles.css" rel="stylesheet" media="screen, print" type="text/css"/>
  </head>
  <body>

    <xsl:call-template name="header"/>

    <div id="content_2">

      <xsl:apply-templates select="xhtml:div[@id = 'menu']"/>

      <div id="Content" class="bulleted">
        <xsl:attribute name="bxexpath"//xhtml:div[@id='Content']</xsl:attribute>
        <xsl:apply-templates select="//xhtml:div[@id = 'Content']"/>
      </div>

      <form action="/cgi-bin/formmail.cgi">
        <input type="hidden" name="your form handlers parameters, etc" value="whatever"/>
        <table border="0">
          <tr>
            <td colspan="4"><strong>Your details</strong></td>
          </tr>
          <tr>
            <td>Name: <span class="required small">(Required)</span></td>
            <td colspan="3"><input type="text" name="Name" size="55"/></td>
          </tr>

... rest of file ....

```

OK now the template's set up. Now we need to set up the pages in Lanya and trick them into using the new template.

1. Create a new 'contact us' page and set up the content how you want it.
2. Create a 'thank you' page and make it hidden in the navigation.
3. Configure your external form handler how you normally would for a static HTML site. Set it up to redirect users back to your 'thank you' page.
4. Go to the your_pub/content/authoring/contact/index_en.html (or wherever it is in your page tree/language) and edit the source. This part is a bit of a hack. An example is below. The main thing is to change the default namespace to the new namespace you created in the first code block above (http://www.my_organisation.com/lenya/contactform) and then make sure that all the html elements have an xhtml: prefix.

5. Reboot Lenya and you're away

```
<?xml version="1.0" encoding="UTF-8"?><html xmlns="http://www.my_organisation.com/lenya/contactform" xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:dcterms="http://purl.org/dc/terms/" xmlns:lenya="http://apache.org/cocoon/lenya/page-envelope/1.0" xmlns:xhtml="http://www.w3.org/1999/xhtml" dc:dummy="FIXME:keepNamespace" dcterms:dummy="FIXME:keepNamespace" lenya:dummy="FIXME:keepNamespace" xhtml:dummy="FIXME:keepNamespace">
  <lenya:meta>
</lenya:meta>
<xhtml:head>
  <xhtml:title>Contact form</xhtml:title>
</xhtml:head>
<xhtml:body>
  <xhtml:p>get in touch with us using the form below.</xhtml:p>

</xhtml:body>
</html>
```