

XSLT Ingest Example: Fill in URPs and UROs

Fill in URPs and UROs

[Start](#) [Previous](#) [Next](#)

This step is actually a replay of the part of the gather XSLT that tries to find an existing match for URIs among the pre-existing foaf:Persons and foaf:Organizations. The main difference is that we use NewPers.xml and NewOrgs.xml in place of Per0.xml and Org0.xml. Figure 14 shows the code for the main XSLT template in fillURPs.xsl. The code for organizations is found in fillUROs.xsl.

```
<xsl:template match="/EduRecords">
  <EduRecords>

    <xsl:for-each select='EduRecord'>

      <xsl:copy>                                0

      <xsl:for-each select='*>

        <xsl:choose>

          <xsl:when test='name() = "personUri" and . = "">          1
            <xsl:variable name='nid' select='../nid'/>
            <xsl:variable name='f' select='../fn'/>
            <xsl:variable name='m' select='../mn'/>
            <xsl:variable name='l' select='../ln'/>

            <xsl:variable name='kuri'
              select='vfx:findMatchingPeople($f, $m, $l,
                                              $nid, $extantPersons)'/>

            <personUri><xsl:value-of select='$kuri[1]'/></personUri>
          </xsl:when>

          <xsl:otherwise>                                2
            <xsl:copy-of select='.'/>
          </xsl:otherwise>

        </xsl:choose>

      </xsl:for-each>

      </xsl:copy>                                3

    </xsl:for-each>
  </EduRecords>
  <xsl:text>
</xsl:text>
</xsl:template>
```

fillURPs.xsl Main Template - Figure 14

- [F14H0] Copy EduRecord as a container.
- [F14H1] When the current tag name matches personUri and the tag is empty, try to find a match to the name/netid peer tags. A match is a sure thing now because of all of the effort that came before.
- [F14H2] Otherwise just deep copy the peer tag data.
- [F14H3] Close out the outer copy operation.

[Start](#) [Previous](#) [Next](#)