```
<?xml version="1.0" encoding="UTF-8"?>
<!-- $Id: MARC21slim2DC.xsl,v 1.1 2003/01/06 08:20:27 adam Exp $ -->
<!DOCTYPE stylesheet>
<xsl:stylesheet version="1.0"
  xmlns:marc="http://www.loc.gov/MARC21/slim"
  xmlns:items="http://www.koha-community.org/items"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:exsl="http://exslt.org/common"
  xmlns:str="http://exslt.org/strings"
  exclude-result-prefixes="marc items str" extension-element-prefixes="exsl">
    <xsl:import href="MARC21slimUtils.xsl"/>
    <xsl:output method = "html" indent="yes" omit-xml-declaration = "yes" encoding="UTF-8"/>

    <xsl:key name="item-by-status" match="items:item" use="items:status"/>

    <xsl:template match="/">
            <xsl:apply-templates/>
    </xsl:template>
    <xsl:template match="marc:record">

        <xsl:variable name="itemcount" select="count(items:items/items:item)"/>

        <!-- Option: Display Alternate Graphic Representation (MARC 880)  -->
        <xsl:variable name="display880" select="boolean(marc:datafield[@tag=880])"/>

    <xsl:variable name="OPACResultsMaxItems" select="number(marc:sysprefs/marc:syspref[@name='OPACResultsMaxItems']+0)"/>
    <xsl:variable name="OPACResultsMaxItemsUnavailable" select="number(marc:sysprefs/marc:syspref[@name='OPACResultsMaxItemsUnavailable']+0)"/>
    <xsl:variable name="OPACResultsUnavailableGroupingBy" select="marc:sysprefs/marc:syspref[@name='OPACResultsUnavailableGroupingBy']"/>
    <xsl:variable name="UseControlNumber" select="marc:sysprefs/marc:syspref[@name='UseControlNumber']"/>
    <xsl:variable name="UseAuthoritiesForTracings" select="marc:sysprefs/marc:syspref[@name='UseAuthoritiesForTracings']"/>
    <xsl:variable name="OPACResultsLibrary" select="marc:sysprefs/marc:syspref[@name='OPACResultsLibrary']"/>
    <xsl:variable name="hidelostitems" select="marc:sysprefs/marc:syspref[@name='hidelostitems']"/>
    <xsl:variable name="DisplayOPACiconsXSLT" select="marc:sysprefs/marc:syspref[@name='DisplayOPACiconsXSLT']"/>
    <xsl:variable name="OPACURLOpenInNewWindow" select="marc:sysprefs/marc:syspref[@name='OPACURLOpenInNewWindow']"/>
    <xsl:variable name="URLLinkText" select="marc:sysprefs/marc:syspref[@name='URLLinkText']"/>
    <xsl:variable name="Show856uAsImage" select="marc:sysprefs/marc:syspref[@name='OPACDisplay856uAsImage']"/>
    <xsl:variable name="AlternateHoldingsField" select="substring(marc:sysprefs/marc:syspref[@name='AlternateHoldingsField'], 1, 3)"/>
    <xsl:variable name="AlternateHoldingsSubfields" select="substring(marc:sysprefs/marc:syspref[@name='AlternateHoldingsField'], 4)"/>
    <xsl:variable name="AlternateHoldingsSeparator" select="marc:sysprefs/marc:syspref[@name='AlternateHoldingsSeparator']"/>
    <xsl:variable name="OPACItemLocation" select="marc:sysprefs/marc:syspref[@name='OPACItemLocation']"/>
    <xsl:variable name="OPACTrackClicks" select="marc:sysprefs/marc:syspref[@name='TrackClicks']"/>
    <xsl:variable name="BiblioDefaultView" select="marc:sysprefs/marc:syspref[@name='BiblioDefaultView']"/>
        <xsl:variable name="leader" select="marc:leader"/>
        <xsl:variable name="leader6" select="substring($leader,7,1)"/>
        <xsl:variable name="leader7" select="substring($leader,8,1)"/>
        <xsl:variable name="leader19" select="substring($leader,20,1)"/>
        <xsl:variable name="biblionumber" select="marc:datafield[@tag=999]/marc:subfield[@code='c']"/>
        <xsl:variable name="isbn" select=
"marc:datafield[@tag=020]/marc:subfield[@code='a']"/>
<xsl:variable name="controlField008" select="marc:controlfield[@tag=008]"/>
<xsl:variable name="typeOf008">
    <xsl:choose>
        <xsl:when test="$leader19='a'">ST</xsl:when>
        <xsl:when test="$leader6='a'">
            <xsl:choose>
                <xsl:when test="$leader7='a' or $leader7='c' or $leader7='d' or
                $leader7='m'">BK</xsl:when>
                <xsl:when test="$leader7='b' or $leader7='i' or $leader7='s'">CR
                </xsl:when>
            </xsl:choose>
        </xsl:when>
        <xsl:when test="$leader6='t'">BK</xsl:when>
        <xsl:when test="$leader6='o' or $leader6='p'">MX</xsl:when>
        <xsl:when test="$leader6='m'">CF</xsl:when>
        <xsl:when test="$leader6='e' or $leader6='f'">MP</xsl:when>
        <xsl:when test="$leader6='g' or $leader6='k' or $leader6='r'">VM
        </xsl:when>
        <xsl:when test="$leader6='i' or $leader6='j'">MU</xsl:when>
        <xsl:when test="$leader6='c' or $leader6='d'">PR</xsl:when>
    </xsl:choose>
</xsl:variable>
<xsl:variable name="controlField008-23" select="substring($controlField008,24,1)"
/>
<xsl:variable name="controlField008-21" select="substring($controlField008,22,1)"
/>
<xsl:variable name="controlField008-22" select="substring($controlField008,23,1)"
/>
<xsl:variable name="controlField008-24" select="substring($controlField008,25,4)"
/>
<xsl:variable name="controlField008-26" select="substring($controlField008,27,1)"
/>
<xsl:variable name="controlField008-29" select="substring($controlField008,30,1)"
/>
<xsl:variable name="controlField008-34" select="substring($controlField008,35,1)"
/>
<xsl:variable name="controlField008-33" select="substring($controlField008,34,1)"
/>
<xsl:variable name="controlField008-30-31" select=
"substring($controlField008,31,2)"/>

<xsl:variable name="physicalDescription">
    <xsl:if test="$typeOf008='CF' and
    marc:controlfield[@tag=007][substring(.,12,1)='a']">
        reformatted digital
    </xsl:if>
    <xsl:if test="$typeOf008='CF' and
    marc:controlfield[@tag=007][substring(.,12,1)='b']">
        digitized microfilm
    </xsl:if>
    <xsl:if test="$typeOf008='CF' and
    marc:controlfield[@tag=007][substring(.,12,1)='d']">
        digitized other analog
    </xsl:if>

    <xsl:variable name="check008-23">
        <xsl:if test="$typeOf008='BK' or $typeOf008='MU' or $typeOf008='CR' or
        $typeOf008='MX'">
            <xsl:value-of select="true()"></xsl:value-of>
        </xsl:if>
    </xsl:variable>
    <xsl:variable name="check008-29">
        <xsl:if test="$typeOf008='MP' or $typeOf008='VM'">
            <xsl:value-of select="true()"></xsl:value-of>
        </xsl:if>
    </xsl:variable>
```

```
            <xsl:choose>
                <xsl:when test="($check008-23 and $controlField008-23='f') or
                ($check008-29 and $controlField008-29='f')">
                    braille
                </xsl:when>
                <xsl:when test="($controlField008-23=' ' and ($leader6='c' or
                $leader6='d')) or (($typeOf008='BK' or $typeOf008='CR') and
                ($controlField008-23=' ' or $controlField008='r'))">
                    print
                </xsl:when>
                <xsl:when test="$leader6 = 'm' or ($check008-23 and
                $controlField008-23='s') or ($check008-29 and $controlField008-29='s')">
                    electronic
                </xsl:when>
                <xsl:when test="($check008-23 and $controlField008-23='b') or
                ($check008-29 and $controlField008-29='b')">
                    microfiche
                </xsl:when>
                <xsl:when test="($check008-23 and $controlField008-23='a') or
                ($check008-29 and $controlField008-29='a')">
                    microfilm
                </xsl:when>
                <xsl:when test="($controlField008-23='d' and ($typeOf008='BK' or
                $typeOf008='CR'))">
                    large print
                </xsl:when>
            </xsl:choose>
<!--
            <xsl:if test="marc:datafield[@tag=130]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
                            <xsl:value-of
                            select="marc:datafield[@tag=130]/marc:subfield[@code='h']"></
                            xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:if test="marc:datafield[@tag=240]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
                            <xsl:value-of
                            select="marc:datafield[@tag=240]/marc:subfield[@code='h']"></
                            xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:if test="marc:datafield[@tag=242]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
                            <xsl:value-of
                            select="marc:datafield[@tag=242]/marc:subfield[@code='h']"></
                            xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:if test="marc:datafield[@tag=245]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
                            <xsl:value-of
                            select="marc:datafield[@tag=245]/marc:subfield[@code='h']"></
                            xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:if test="marc:datafield[@tag=246]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
```

```
                            <xsl:value-of select="marc:datafield[@tag=246]/marc:subfield[@code='h']"></xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:if test="marc:datafield[@tag=730]/marc:subfield[@code='h']">
                    <xsl:call-template name="chopBrackets">
                        <xsl:with-param name="chopString">
                            <xsl:value-of select="marc:datafield[@tag=730]/marc:subfield[@code='h']"></xsl:value-of>
                        </xsl:with-param>
                    </xsl:call-template>
            </xsl:if>
            <xsl:for-each select="marc:datafield[@tag=256]/marc:subfield[@code='a']">
                    <xsl:value-of select="."></xsl:value-of>
            </xsl:for-each>
            <xsl:for-each select="marc:controlfield[@tag=007][substring(text(),1,1)='c']">
                <xsl:choose>
                    <xsl:when test="substring(text(),14,1)='a'">
                        access
                    </xsl:when>
                    <xsl:when test="substring(text(),14,1)='p'">
                        preservation
                    </xsl:when>
                    <xsl:when test="substring(text(),14,1)='r'">
                        replacement
                    </xsl:when>
                </xsl:choose>
            </xsl:for-each>
-->
<!-- Modified for authcat: Material type section removed -->
<!--
            <xsl:for-each select="marc:datafield[@tag=856]/marc:subfield[@code='q'][string-length(.)>1]">
                    <xsl:value-of select="."></xsl:value-of>
            </xsl:for-each>
            <xsl:for-each select="marc:datafield[@tag=300]">
                    <xsl:call-template name="subfieldSelect">
                        <xsl:with-param name="codes">abce</xsl:with-param>
                    </xsl:call-template>
            </xsl:for-each>
-->
        </xsl:variable>

        <!-- Title Statement: Alternate Graphic Representation (MARC 880) -->
        <xsl:if test="$display880">
           <xsl:call-template name="m880Select">
              <xsl:with-param name="basetags">245</xsl:with-param>
              <xsl:with-param name="codes">abhfgknps</xsl:with-param>
              <xsl:with-param name="bibno"><xsl:value-of  select="$biblionumber"/></xsl:with-param>
           </xsl:call-template>
        </xsl:if>

    <a>
        <xsl:attribute name="href">
            <xsl:call-template name="buildBiblioDefaultViewURL">
                <xsl:with-param name="BiblioDefaultView">
                    <xsl:value-of select="$BiblioDefaultView"/>
                </xsl:with-param>
            </xsl:call-template>
            <xsl:value-of select="str:encode-uri($biblionumber, true())"/>
        </xsl:attribute>
```

```
            <xsl:attribute name="class"><xsl:value-of select="'title'" /></xsl:attribute>

            <xsl:if test="marc:datafield[@tag=245]">
                <xsl:for-each select="marc:datafield[@tag=245]">
                    <xsl:call-template name="subfieldSelect">
                        <xsl:with-param name="codes">a</xsl:with-param>
                    </xsl:call-template>
                    <xsl:text> </xsl:text>
                    <!-- 13381 add additional subfields-->
                    <!-- bz 17625 adding subfields f and g -->
                    <xsl:for-each select="marc:subfield[contains('bcfghknps', @code)]">
                        <xsl:choose>
                            <xsl:when test="@code='h'">
                                <!--  13381 Span class around subfield h so it can be suppressed via css -->
                                <span class="title_medium"><xsl:apply-templates/> <xsl:text> </xsl:text> </span>
                            </xsl:when>
                            <xsl:when test="@code='c'">
                                <!--  13381 Span class around subfield c so it can be suppressed via css -->
                                <span class="title_resp_stmt"><xsl:apply-templates/> <xsl:text> </xsl:text> </span>
                            </xsl:when>
                            <xsl:otherwise>
                                <xsl:apply-templates/>
                                <xsl:text> </xsl:text>
                            </xsl:otherwise>
                        </xsl:choose>
                    </xsl:for-each>
                </xsl:for-each>
            </xsl:if>
        </a>

        <!-- OpenURL -->
        <xsl:variable name="OPACShowOpenURL" select="marc:sysprefs/marc:syspref[@name='OPACShowOpenURL']" />
        <xsl:variable name="OpenURLImageLocation" select="marc:sysprefs/marc:syspref[@name='OpenURLImageLocation']" />
        <xsl:variable name="OpenURLText" select="marc:sysprefs/marc:syspref[@name='OpenURLText']" />
        <xsl:variable name="OpenURLResolverURL" select="marc:variables/marc:variable[@name='OpenURLResolverURL']" />

        <xsl:if test="$OPACShowOpenURL = 1 and $OpenURLResolverURL != ''">
          <xsl:variable name="openurltext">
            <xsl:choose>
              <xsl:when test="$OpenURLText != ''">
                <xsl:value-of select="$OpenURLText" />
              </xsl:when>
              <xsl:otherwise>
                <xsl:text>OpenURL</xsl:text>
              </xsl:otherwise>
            </xsl:choose>
          </xsl:variable>

          <div class="results_summary"><a>
            <xsl:attribute name="href">
              <xsl:value-of select="$OpenURLResolverURL" />
            </xsl:attribute>
            <xsl:attribute name="title">
              <xsl:value-of select="$openurltext" />
            </xsl:attribute>
            <xsl:attribute name="class">
              <xsl:text>OpenURL</xsl:text>
            </xsl:attribute>
            <xsl:if test="$OPACURLOpenInNewWindow='1'">
```

```
     <xsl:attribute name="target">
      <xsl:text>_blank</xsl:text>
     </xsl:attribute>
    </xsl:if>
    <xsl:choose>
     <xsl:when test="$OpenURLImageLocation != ''">
      <img>
       <xsl:attribute name="src">
        <xsl:value-of select="$OpenURLImageLocation" />
       </xsl:attribute>
      </img>
     </xsl:when>
     <xsl:otherwise>
      <xsl:value-of select="$openurltext" />
     </xsl:otherwise>
    </xsl:choose>
   </a></div>
  </xsl:if>
  <!-- End of OpenURL -->

  <!-- Author Statement: Alternate Graphic Representation (MARC 880) -->
  <xsl:if test="$display880">
   <xsl:call-template name="m880Select">
   <xsl:with-param name="basetags">100,110,111,700,710,711</xsl:with-param>
   <xsl:with-param name="codes">abc</xsl:with-param>
   </xsl:call-template>
  </xsl:if>

  <xsl:choose>
  <xsl:when test="marc:datafield[@tag=100] or marc:datafield[@tag=110] or marc:datafield[@tag=111] or marc:datafield[@tag=700] or marc:datafield[@tag=710] or marc:datafield[@tag=711]">
<!-- Modified for authcat:  "MedStar authors:-->
    <span class="byAuthor">MedStar authors: </span>
    <ul class="author resource_list">
    <!-- #13383 -->
    <xsl:for-each select="marc:datafield[(@tag=100 or @tag=700 or @tag=110 or @tag=710 or @tag=111 or @tag=711) and @ind1!='z']">
      <li>
      <xsl:call-template name="chopPunctuation">
        <xsl:with-param name="chopString">
          <xsl:call-template name="subfieldSelect">
            <xsl:with-param name="codes">
              <xsl:choose>
                <!-- #13383 include subfield e for field 111  -->
                <xsl:when test="@tag=111 or @tag=711">a</xsl:when>
                <xsl:when test="@tag=110 or @tag=710">a</xsl:when>
                <xsl:otherwise>a</xsl:otherwise>
              </xsl:choose>
            </xsl:with-param>
          </xsl:call-template>
        </xsl:with-param>
        <xsl:with-param name="punctuation">
          <xsl:text>:,;/ </xsl:text>
        </xsl:with-param>
      </xsl:call-template>
      <!-- Display title portion for 110 and 710 fields -->
<!--
      <xsl:if test="(@tag=110 or @tag=710) and boolean(marc:subfield[@code='c' or @code='d' or @code='n' or @code='t'])">
        <span class="titleportion">
        <xsl:choose>
          <xsl:when test="marc:subfield[@code='c' or @code='d' or @code='n'][not(marc:subfield[@code='t'])]"><xsl:text> </xsl:text></xsl:when>
          <xsl:otherwise><xsl:text>. </xsl:text></xsl:otherwise>
        </xsl:choose>
```

```
                <xsl:call-template name="chopPunctuation">
                    <xsl:with-param name="chopString">
                    <xsl:call-template name="subfieldSelect">
                        <xsl:with-param name="codes">cdnt</xsl:with-param>
                    </xsl:call-template>
                    </xsl:with-param>
                </xsl:call-template>
                </span>
            </xsl:if>
-->
            <!-- Display title portion for 111 and 711 fields -->
            <xsl:if test="(@tag=111 or @tag=711) and boolean(marc:subfield[@code='c' or @code='d' or @code='g' or @code='n' or @code='t'])">
                    <span class="titleportion">
                    <xsl:choose>
                        <xsl:when test="marc:subfield[@code='c' or @code='d' or @code='g' or @code='n'][not(marc:subfield[@code='t'])]"><xsl:text> </xsl:text></xsl:when>
                        <xsl:otherwise><xsl:text>. </xsl:text></xsl:otherwise>
                    </xsl:choose>

                    <xsl:call-template name="chopPunctuation">
                        <xsl:with-param name="chopString">
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">cdgnt</xsl:with-param>
                        </xsl:call-template>
                        </xsl:with-param>
                    </xsl:call-template>
                    </span>
            </xsl:if>
            <!-- Display dates for 100 and 700 fields -->
<!--
            <xsl:if test="(@tag=100 or @tag=700) and marc:subfield[@code='d']">
                <span class="authordates">
                <xsl:text>, </xsl:text>
                <xsl:call-template name="chopPunctuation">
                    <xsl:with-param name="chopString">
                        <xsl:call-template name="subfieldSelect">
                           <xsl:with-param name="codes">d</xsl:with-param>
                        </xsl:call-template>
                    </xsl:with-param>
                </xsl:call-template>
                </span>
            </xsl:if>
-->
            <!-- Display title portion for 100 and 700 fields -->
            <xsl:if test="@tag=700 and marc:subfield[@code='t']">
                <span class="titleportion">
                <xsl:text>. </xsl:text>
                <xsl:call-template name="chopPunctuation">
                    <xsl:with-param name="chopString">
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">t</xsl:with-param>
                        </xsl:call-template>
                    </xsl:with-param>
                </xsl:call-template>
                </span>
            </xsl:if>
            <!-- Display relators for 1XX and 7XX fields (code e replaced with 4 -->
                        <xsl:if test="marc:subfield[@code='4' or @code='4'][not(parent::*[@tag=111])] or (self::*[@tag=111] and marc:subfield[@code='4' or @code='j'][. != ''])">
                <span class="relatorcode">
                    <xsl:text> [</xsl:text>
                    <xsl:choose>
                        <xsl:when test="@tag=111 or @tag=711">
                            <xsl:choose>
```

```
                                    <!-- Prefer j over 4 for 111 and 711 -->

                                    <xsl:when test="marc:subfield[@code='j']">
                                        <xsl:for-each select="marc:subfield[@code='j']">
                                            <xsl:value-of select="."/>
                                            <xsl:if test="position() != last()">, </xsl:if>
                                        </xsl:for-each>
                                    </xsl:when>
                                    <xsl:otherwise>
                                        <xsl:for-each select="marc:subfield[@code=4]">
                                            <xsl:value-of select="."/>
                                            <xsl:if test="position() != last()">, </xsl:if>
                                        </xsl:for-each>
                                    </xsl:otherwise>
                                </xsl:choose>
                            </xsl:when>
                            <!-- Prefer e over 4 on 100 and 110 -->

                            <xsl:when test="marc:subfield[@code='e']">
                                <xsl:for-each select="marc:subfield[@code='4'][not(@tag=111) or not(@tag=711)]">
                                    <xsl:value-of select="."/>
                                    <xsl:if test="position() != last()">, </xsl:if>
                                </xsl:for-each>
                            </xsl:when>
                            <xsl:otherwise>
                                <xsl:for-each select="marc:subfield[@code=4]">
                                    <xsl:value-of select="."/>
                                    <xsl:if test="position() != last()">, </xsl:if>
                                </xsl:for-each>
                            </xsl:otherwise>
                        </xsl:choose>
                        <xsl:text>]</xsl:text>
                    </span>
                </xsl:if>


                </li>
            </xsl:for-each>
            </ul>
        </xsl:when>
        </xsl:choose>

        <xsl:call-template name="show-series">
            <xsl:with-param name="searchurl">/cgi-bin/koha/opac-search.pl</xsl:with-param>
            <xsl:with-param name="UseControlNumber" select="$UseControlNumber"/>
            <xsl:with-param name="UseAuthoritiesForTracings" select="$UseAuthoritiesForTracings"/>
        </xsl:call-template>

        <xsl:if test="marc:datafield[@tag=250]">
        <div class="results_summary edition">
        <span class="label">Edition: </span>
                <xsl:for-each select="marc:datafield[@tag=250]">
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">ab</xsl:with-param>
                        </xsl:call-template>
                </xsl:for-each>
        </div>
        </xsl:if>

        <xsl:if test="marc:datafield[@tag=773]">
            <xsl:for-each select="marc:datafield[@tag=773]">
                <xsl:if test="marc:subfield[@code='t']">
        <div class="results_summary source">
        <span class="label">Source: </span>
                <xsl:value-of select="marc:subfield[@code='t']"/>
```

```
    </div>
            </xsl:if>
        </xsl:for-each>
    </xsl:if>

<xsl:if test="$DisplayOPACiconsXSLT!='0'">
    <div class="results_summary type">
    <xsl:if test="$typeOf008!=''">
    <span class="results_material_type">
        <span class="label">Material type: </span>
            <xsl:choose>
                <xsl:when test="$leader19='a'"><img src="/opac-tmpl/lib/famfamfam/ST.png" alt="Set" class="materialtype mt_icon_ST"/> Set</xsl:when>
                <xsl:when test="$leader6='a'">
                    <xsl:choose>
                        <xsl:when test="$leader7='c' or $leader7='d' or $leader7='m'"><img src="/opac-tmpl/lib/famfamfam/BK.png" alt="Text" class="materialtype mt_icon_BK"/> Text</xsl:when>
                        <xsl:when test="$leader7='i' or $leader7='s'"><img src="/opac-tmpl/lib/famfamfam/SE.png" alt="Continuing resource" class="materialtype mt_icon_SE"/> Continuing resource</xsl:when>
                        <xsl:when test="$leader7='a' or $leader7='b'"><img src="/opac-tmpl/lib/famfamfam/AR.png" alt="Article" class="materialtype mt_icon_AR"/> Article</xsl:when>
                    </xsl:choose>
                </xsl:when>
                <xsl:when test="$leader6='t'"><img src="/opac-tmpl/lib/famfamfam/BK.png" alt="Text" class="materialtype mt_icon_BK"/> Text</xsl:when>
                <xsl:when test="$leader6='o'"><img src="/opac-tmpl/lib/famfamfam/MM.png" alt="Kit" class="materialtype mt_icon_MM"/> Kit</xsl:when>
                <xsl:when test="$leader6='p'"><img src="/opac-tmpl/lib/famfamfam/MM.png" alt="Mixed materials" class="materialtype mt_icon_MM"/>Mixed materials</xsl:when>
                <xsl:when test="$leader6='m'"><img src="/opac-tmpl/lib/famfamfam/CF.png" alt="Computer file" class="materialtype mt_icon_CF"/> Computer file</xsl:when>
                <xsl:when test="$leader6='e' or $leader6='f'"><img src="/opac-tmpl/lib/famfamfam/MP.png" alt="Map" class="materialtype mt_icon_MP"/> Map</xsl:when>
                <xsl:when test="$leader6='g'"><img src="/opac-tmpl/lib/famfamfam/VM.png" alt="Film" class="materialtype mt_icon_VM"/> Film</xsl:when>
                <xsl:when test="$leader6='k'"><img src="/opac-tmpl/lib/famfamfam/GR.png" alt="Picture" class="materialtype mt_icon_GR"/> Picture</xsl:when>
                <xsl:when test="$leader6='r'"><img src="/opac-tmpl/lib/famfamfam/OB.png" alt="Object" class="materialtype mt_icon_OB"/> Object</xsl:when>
                <xsl:when test="$leader6='c' or $leader6='d'"><img src="/opac-tmpl/lib/famfamfam/PR.png" alt="Score" class="materialtype mt_icon_PR"/> Score</xsl:when>
                <xsl:when test="$leader6='i'"><img src="/opac-tmpl/lib/famfamfam/MU.png" alt="Sound" class="materialtype mt_icon_MU"/> Sound</xsl:when>
                <xsl:when test="$leader6='j'"><img src="/opac-tmpl/lib/famfamfam/PR.png" alt="Music" class="materialtype mt_icon_PR"/> Music</xsl:when>
            </xsl:choose>
    </span>
    </xsl:if>
    <xsl:if test="string-length(normalize-space($physicalDescription))" >
        <span class="results_format">
            <span class="label">; Format: </span><xsl:copy-of select="$physicalDescription"></xsl:copy-of>
        </span>
    </xsl:if>

        <xsl:if test="$controlField008-21 or $controlField008-22 or $controlField008-24 or $controlField008-26 or $controlField008-29 or $controlField008-34 or $controlField008-33 or $controlField008-30-31 or $controlField008-33" >

        <xsl:if test="$typeOf008='CR'">
```

```
<span class="results_typeofcontinuing">
<xsl:if test="$controlField008-21 and $controlField008-21 !='|' and $controlField008-21 !=' '">
<span class="label">; Type of continuing resource: </span>
</xsl:if>
    <xsl:choose>
        <xsl:when test="$controlField008-21='d'">
            <img src="/opac-tmpl/lib/famfamfam/database.png" alt="database" class="format"/>
        </xsl:when>
        <xsl:when test="$controlField008-21='l'">
            loose-leaf
        </xsl:when>
        <xsl:when test="$controlField008-21='m'">
            series
        </xsl:when>
        <xsl:when test="$controlField008-21='n'">
            newspaper
        </xsl:when>
        <xsl:when test="$controlField008-21='p'">
            periodical
        </xsl:when>
        <xsl:when test="$controlField008-21='w'">
             <img src="/opac-tmpl/lib/famfamfam/world_link.png" alt="web site" class="format"/>
        </xsl:when>
    </xsl:choose>
</span>
</xsl:if>
<xsl:if test="$typeOf008='BK' or $typeOf008='CR'">
<xsl:if test="contains($controlField008-24,'abcdefghijklmnopqrstvwxyz')">
<span class="results_natureofcontents">
<span class="label">; Nature of contents: </span>
    <xsl:choose>
        <xsl:when test="contains($controlField008-24,'a')">
            abstract or summary
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'b')">
    bibliography
             <img src="/opac-tmpl/lib/famfamfam/text_list_bullets.png" alt="bibliography" class="natureofcontents"/>
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'c')">
            catalog
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'d')">
            dictionary
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'e')">
            encyclopedia
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'f')">
            handbook
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'g')">
            legal article
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'i')">
            index
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'k')">
            discography
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'l')">
            legislation
        </xsl:when>
        <xsl:when test="contains($controlField008-24,'m')">
```

```
                        theses
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'n')">
                        survey of literature
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'o')">
                        review
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'p')">
                        programmed text
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'q')">
                        filmography
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'r')">
                        directory
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'s')">
                        statistics
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'t')">
                         <img src="/opac-tmpl/lib/famfamfam/report.png" alt="technical report" class="natureofcontents"/>
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'v')">
                        legal case and case notes
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'w')">
                        law report or digest
                    </xsl:when>
                    <xsl:when test="contains($controlField008-24,'z')">
                        treaty
                    </xsl:when>
                </xsl:choose>
                <xsl:choose>
                    <xsl:when test="$controlField008-29='1'">
                        conference publication
                    </xsl:when>
                </xsl:choose>
            </span>
            </xsl:if>
            </xsl:if>
            <xsl:if test="$typeOf008='CF'">
            <span class="results_typeofcomp">
                <xsl:if test="$controlField008-26='a' or $controlField008-26='e' or $controlField008-26='f' or $controlField008-26='g'">
                <span class="label">; Type of computer file: </span>
                </xsl:if>
                <xsl:choose>
                    <xsl:when test="$controlField008-26='a'">
                        numeric data
                    </xsl:when>
                    <xsl:when test="$controlField008-26='e'">
                         <img src="/opac-tmpl/lib/famfamfam/database.png" alt="database" class="format"/>
                    </xsl:when>
                    <xsl:when test="$controlField008-26='f'">
                         <img src="/opac-tmpl/lib/famfamfam/font.png" alt="font" class="format"/>
                    </xsl:when>
                    <xsl:when test="$controlField008-26='g'">
                         <img src="/opac-tmpl/lib/famfamfam/controller.png" alt="game" class="format"/>
                    </xsl:when>
                </xsl:choose>
            </span>
            </xsl:if>
```

```
        <xsl:if test="$typeOf008='BK'">
        <span class="results_contents_literary">
            <xsl:if test="(substring($controlField008,25,1)='j') or (substring($controlField008,25,1)='1') or ($controlField008-34='a' or $controlField008-34='b' or $controlField008-34='c' or $controlField008-34='d')">
            <span class="label">; Nature of contents: </span>
            </xsl:if>
            <xsl:if test="substring($controlField008,25,1)='j'">
                patent
            </xsl:if>
            <xsl:if test="substring($controlField008,31,1)='1'">
                festschrift
            </xsl:if>
            <xsl:if test="$controlField008-34='a' or $controlField008-34='b' or $controlField008-34='c' or $controlField008-34='d'">
                 <img src="/opac-tmpl/lib/famfamfam/user.png" alt="biography" class="natureofcontents"/>
            </xsl:if>

            <xsl:if test="$controlField008-33 and $controlField008-33!='|' and $controlField008-33!='u' and $controlField008-33!=' '">
            <span class="label">; Literary form: </span>
            </xsl:if>
            <xsl:choose>
                <xsl:when test="$controlField008-33='0'">
                    Not fiction
                </xsl:when>
                <xsl:when test="$controlField008-33='1'">
                    Fiction
                </xsl:when>
                <xsl:when test="$controlField008-33='d'">
                    Dramas
                </xsl:when>
                <xsl:when test="$controlField008-33='e'">
                    Essays
                </xsl:when>
                <xsl:when test="$controlField008-33='f'">
                    Novels
                </xsl:when>
                <xsl:when test="$controlField008-33='h'">
                    Humor, satires, etc.
                </xsl:when>
                <xsl:when test="$controlField008-33='i'">
                    Letters
                </xsl:when>
                <xsl:when test="$controlField008-33='j'">
                    Short stories
                </xsl:when>
                <xsl:when test="$controlField008-33='m'">
                    Mixed forms
                </xsl:when>
                <xsl:when test="$controlField008-33='p'">
                    Poetry
                </xsl:when>
                <xsl:when test="$controlField008-33='s'">
                    Speeches
                </xsl:when>
            </xsl:choose>
        </span>
        </xsl:if>
        <xsl:if test="$typeOf008='MU' and $controlField008-30-31 and $controlField008-30-31!='||' and $controlField008-30-31!='  '">
        <span class="results_literaryform">
            <span class="label">; Literary form: </span> <!-- Literary text for sound recordings -->
            <xsl:if test="contains($controlField008-30-31,'b')">
```

```
                    biography
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'c')">
                    conference publication
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'d')">
                    drama
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'e')">
                    essay
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'f')">
                    fiction
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'o')">
                    folktale
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'h')">
                    history
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'k')">
                    humor, satire
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'m')">
                    memoir
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'p')">
                    poetry
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'r')">
                    rehearsal
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'g')">
                    reporting
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'s')">
                    sound
                </xsl:if>
                <xsl:if test="contains($controlField008-30-31,'l')">
                    speech
                </xsl:if>
            </span>
            </xsl:if>
            <xsl:if test="$typeOf008='VM'">
            <span class="results_typeofvisual">
                <span class="label">; Type of visual material: </span>
                <xsl:choose>
                    <xsl:when test="$controlField008-33='a'">
                        art original
                    </xsl:when>
                    <xsl:when test="$controlField008-33='b'">
                        kit
                    </xsl:when>
                    <xsl:when test="$controlField008-33='c'">
                        art reproduction
                    </xsl:when>
                    <xsl:when test="$controlField008-33='d'">
                        diorama
                    </xsl:when>
                    <xsl:when test="$controlField008-33='f'">
                        filmstrip
                    </xsl:when>
                    <xsl:when test="$controlField008-33='g'">
                        game
                    </xsl:when>
                    <xsl:when test="$controlField008-33='i'">
                        picture
                    </xsl:when>
                    <xsl:when test="$controlField008-33='k'">
                        graphic
                    </xsl:when>
                    <xsl:when test="$controlField008-33='l'">
                        technical drawing
                    </xsl:when>
                    <xsl:when test="$controlField008-33='m'">
                        motion picture
                    </xsl:when>
                    <xsl:when test="$controlField008-33='n'">
                        chart
                    </xsl:when>
                    <xsl:when test="$controlField008-33='o'">
                        flash card
                    </xsl:when>
                    <xsl:when test="$controlField008-33='p'">
                        microscope slide
                    </xsl:when>
                    <xsl:when test="$controlField008-33='q' or marc:controlfield[@tag=007][substring(text(),1,1)='a'][substring(text(),2,1)='q']">
                        model
                    </xsl:when>
                    <xsl:when test="$controlField008-33='r'">
                        realia
                    </xsl:when>
                    <xsl:when test="$controlField008-33='s'">
                        slide
                    </xsl:when>
                    <xsl:when test="$controlField008-33='t'">
                        transparency
                    </xsl:when>
                    <xsl:when test="$controlField008-33='v'">
                        videorecording
                    </xsl:when>
                    <xsl:when test="$controlField008-33='w'">
                        toy
                    </xsl:when>
                </xsl:choose>
                </span>
            </xsl:if>
            </xsl:if>

        <xsl:if test="($typeOf008='BK' or $typeOf008='CF' or $typeOf008='MU' or $typeOf008='VM') and ($controlField008-22='a' or $controlField008-22='b' or $controlField008-22='c' or $controlField008-22='d' or $controlField008-22='e' or $controlField008-22='g' or $controlField008-22='j' or $controlField008-22='f')">
            <span class="results_audience">
            <span class="label">; Audience: </span>
            <xsl:choose>
                <xsl:when test="$controlField008-22='a'">
                 Preschool;
                </xsl:when>
                <xsl:when test="$controlField008-22='b'">
                 Primary;
                </xsl:when>
                <xsl:when test="$controlField008-22='c'">
                 Pre-adolescent;
                </xsl:when>
                <xsl:when test="$controlField008-22='d'">
                 Adolescent;
                </xsl:when>
                <xsl:when test="$controlField008-22='e'">
                 Adult;
                </xsl:when>
                <xsl:when test="$controlField008-22='g'">
```

```
             General;
            </xsl:when>
            <xsl:when test="$controlField008-22='j'">
             Juvenile;
            </xsl:when>
            <xsl:when test="$controlField008-22='f'">
             Specialized;
            </xsl:when>
        </xsl:choose>
        </span>
    </xsl:if>
<xsl:text> </xsl:text> <!-- added blank space to fix font display problem, see Bug 3671 -->
</div>
</xsl:if>

    <xsl:call-template name="show-lang-041"/>

    <!-- Publisher Statement: Alternate Graphic Representation (MARC 880) -->
    <xsl:if test="$display880">
      <xsl:call-template name="m880Select">
        <xsl:with-param name="basetags">260</xsl:with-param>
        <xsl:with-param name="codes">abcg</xsl:with-param>
        <xsl:with-param name="class">results_summary publisher</xsl:with-param>
        <xsl:with-param name="label">Publication details: </xsl:with-param>
      </xsl:call-template>
    </xsl:if>


    <!-- PMID -->
<!-- Modified for authcat: changed tag 099-->
    <xsl:if test="marc:datafield[@tag=099]">
         <div class="results_summary publisher_number ">
            <span class="label">PMID: </span>
            <ul class="resource_list">
                <xsl:for-each select="marc:datafield[@tag=099]">
                    <li>
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">a</xsl:with-param>
                        </xsl:call-template>
                    </li>
                </xsl:for-each>
            </ul>
        </div>
    </xsl:if>


    <!-- Publisher info and RDA related info from tags 260, 264 -->
    <xsl:choose>
        <xsl:when test="marc:datafield[@tag=264]">
            <xsl:call-template name="showRDAtag264"/>
        </xsl:when>
        <xsl:when test="marc:datafield[@tag=260]">
<!-- Modified for authcat:  label changed to year-->
            <div class="results_summary publisher"><span class="label">Year: </span>
                <xsl:for-each select="marc:datafield[@tag=260]">
<!--
                    <xsl:if test="marc:subfield[@code='a']">
                        <span class="publisher_place" property="location">
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">a</xsl:with-param>
                            </xsl:call-template>
                        </span>
                    </xsl:if>
                    <xsl:text> </xsl:text>
                    <xsl:if test="marc:subfield[@code='b']">
                        <span property="name" class="publisher_name">
```

```
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">b</xsl:with-param>
                            </xsl:call-template>
                        </span>
                    </xsl:if>
                    <xsl:text> </xsl:text>
-->
                    <span property="datePublished" class="publisher_date">
                        <xsl:call-template name="chopPunctuation">
                            <xsl:with-param name="chopString">
                                    <xsl:call-template name="subfieldSelect">
                                        <xsl:with-param name="codes">c</xsl:with-param>
                                    </xsl:call-template>
                            </xsl:with-param>
                        </xsl:call-template>
                    </span>
<!-- semi-colon taken out -->
                    <xsl:choose><xsl:when test="position()=last()"><xsl:text></xsl:text></xsl:when><xsl:otherwise><xsl:text></xsl:text></xsl:otherwise></xsl:choose>
                </xsl:for-each>
                <xsl:if test="marc:datafield[@tag=264]">
                    <xsl:text>; </xsl:text>
                    <xsl:call-template name="showRDAtag264"/>
                </xsl:if>
            </div>
        </xsl:when>
    </xsl:choose>

 <!-- citation -->
<!-- Modified for authcat: new tag -->
    <xsl:if test="marc:datafield[@tag=251]">
         <div class="results_summary ">
            <span class="label">Citation: </span>
            <ul class="resource_list">
                <xsl:for-each select="marc:datafield[@tag=251]">
                    <li>
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">a</xsl:with-param>
                        </xsl:call-template>
                    </li>
                </xsl:for-each>
            </ul>
        </div>
    </xsl:if>

    <!-- institution -->
<!-- Modified for authcat: changed tag -->
    <xsl:if test="marc:datafield[@tag=651]">
         <div class="results_summary ">
            <span class="label">Institution: </span>
            <ul class="resource_list">
                <xsl:for-each select="marc:datafield[@tag=651]">
                    <li>
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">a</xsl:with-param>
                        </xsl:call-template>
                    </li>
                </xsl:for-each>
            </ul>
        </div>
    </xsl:if>

    <!-- department -->
<!-- Modified for authcat: changed tag -->
    <xsl:if test="marc:datafield[@tag=656]">
         <div class="results_summary ">
```

```
                <span class="label">Department: </span>
                <ul class="resource_list">
                    <xsl:for-each select="marc:datafield[@tag=656]">
                        <li>
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">a</xsl:with-param>
                            </xsl:call-template>
                        </li>
                    </xsl:for-each>
                </ul>
            </div>
        </xsl:if>


        <!-- Publication -->
    <!-- Modified for authcat: changed tag -->
        <xsl:if test="marc:datafield[@tag=657]">
             <div class="results_summary ">
                <span class="label">Medline publication type: </span>
                <ul class="resource_list">
                    <xsl:for-each select="marc:datafield[@tag=657]">
                        <li>
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">a</xsl:with-param>
                            </xsl:call-template>
                        </li>
                    </xsl:for-each>
                </ul>
            </div>
        </xsl:if>


        <!-- Article type -->
    <!-- Modified for authcat: changed tag -->
        <xsl:if test="marc:datafield[@tag=942]">
             <div class="results_summary ">
                <span class="label">Form of publication: </span>
                <ul class="resource_list">
                    <xsl:for-each select="marc:datafield[@tag=942]">
                        <li>
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">c</xsl:with-param>
                            </xsl:call-template>
                        </li>
                    </xsl:for-each>
                </ul>
            </div>
        </xsl:if>


        <!-- AU -->
    <!-- Modified for authcat: changed tag -->
        <xsl:if test="marc:datafield[@tag=790]">
             <div class="results_summary ">
                <span class="label">All authors: </span>
                <ul class="resource_list">
                    <xsl:for-each select="marc:datafield[@tag=790]">
                        <li>
                            <xsl:call-template name="subfieldSelect">
                                <xsl:with-param name="codes">a</xsl:with-param>
                            </xsl:call-template>
                        </li>
                    </xsl:for-each>
                </ul>
            </div>
        </xsl:if>

```

```
<!-- Dissertation note -->
<xsl:if test="marc:datafield[@tag=502]">
    <div class="results_summary diss_note">
        <span class="label">Dissertation note: </span>
        <xsl:for-each select="marc:datafield[@tag=502]">
            <xsl:call-template name="subfieldSelect">
                <xsl:with-param name="codes">abcdgo</xsl:with-param>
            </xsl:call-template>
        </xsl:for-each>
        <xsl:choose><xsl:when test="position()=last()"
        ><xsl:text></xsl:text></xsl:when><xsl:otherwise><xsl:text>
        </xsl:text></xsl:otherwise></xsl:choose>
    </div>
</xsl:if>

<!-- Other Title  Statement: Alternate Graphic Representation (MARC 880) -->
<xsl:if test="$display880">
   <xsl:call-template name="m880Select">
   <!-- Modified for authcat:  -->
     <xsl:with-param name="basetags">099</xsl:with-param>
     <xsl:with-param name="codes">ab</xsl:with-param>
     <xsl:with-param name="class">results_summary other_title</xsl:with-param>
     <xsl:with-param name="label">PMID: </xsl:with-param>
   </xsl:call-template>
</xsl:if>

<xsl:if test="marc:datafield[@tag=246]">
<div class="results_summary other_title">
<span class="label">Other title: </span>
    <ul class="resource_list">
        <xsl:for-each select="marc:datafield[@tag=246]">
            <li>
                <xsl:call-template name="subfieldSelect">
                    <xsl:with-param name="codes">ab</xsl:with-param>
                </xsl:call-template>
            </li>
        </xsl:for-each>
    </ul>
</div>
</xsl:if>
<xsl:if test="marc:datafield[@tag=242]">
<div class="results_summary translated_title">
<span class="label">Title translated: </span>
        <xsl:for-each select="marc:datafield[@tag=242]">
                <xsl:call-template name="subfieldSelect">
                    <xsl:with-param name="codes">abh</xsl:with-param>
                </xsl:call-template>
                <xsl:choose><xsl:when test="position()=last()"><xsl:text>.
                </xsl:text></xsl:when><xsl:otherwise><xsl:text>;
                </xsl:text></xsl:otherwise></xsl:choose>
        </xsl:for-each>
</div>
</xsl:if>
<xsl:if test="marc:datafield[@tag=856]">
<!-- Modified for authcat: DOI -->
     <div class="results_summary online_resources">
           <span class="label">DOI: </span>
                <ul class="resource_list">
                        <xsl:for-each select="marc:datafield[@tag=856]">
                        <xsl:variable name="SubqText"><xsl:value-of select=
                        "marc:subfield[@code='q']"/></xsl:variable>
                        <xsl:if test="$OPACURLOpenInNewWindow='0'">
              <li><a>
              <xsl:choose>
                <xsl:when test="$OPACTrackClicks='track'">
                  <xsl:attribute name="href">/cgi-bin/koha/tracklinks.pl?uri=
                  <xsl:value-of select="str:encode-uri(marc:subfield[@code='u'],
                  true())"/>&amp;biblionumber=<xsl:value-of select="$biblionumber"/>
                  </xsl:attribute>
            </xsl:when>
                        <xsl:when test="$OPACTrackClicks='anonymous'">
                  <xsl:attribute name="href">/cgi-bin/koha/tracklinks.pl?uri=
                  <xsl:value-of select="str:encode-uri(marc:subfield[@code='u'],
                  true())"/>&amp;biblionumber=<xsl:value-of select="$biblionumber"/>
                  </xsl:attribute>
            </xsl:when>
            <xsl:otherwise>
                <xsl:attribute name="href">
                    <xsl:if test="not(contains(marc:subfield[@code='u'],'://'))">
                        <xsl:choose>
                            <xsl:when test="@ind1=7">
                                <xsl:value-of select="marc:subfield[@code='2']"/>
                                <xsl:text>://</xsl:text>
                            </xsl:when>
                            <xsl:when test="@ind1=1">
                                <xsl:text>ftp://</xsl:text>
                            </xsl:when>
                            <xsl:otherwise>
                                <xsl:text>http://</xsl:text>
                            </xsl:otherwise>
                        </xsl:choose>
                    </xsl:if>
                    <xsl:value-of select="marc:subfield[@code='u']"/>
                </xsl:attribute>
            </xsl:otherwise>
              </xsl:choose>
                                <xsl:choose>
                                 <xsl:when test="($Show856uAsImage='Results' or
                                 $Show856uAsImage='Both') and
                                 (substring($SubqText,1,6)='image/' or
                                 $SubqText='img' or $SubqText='bmp' or
                                 $SubqText='cod' or $SubqText='gif' or
                                 $SubqText='ief' or $SubqText='jpe' or
                                 $SubqText='jpeg' or $SubqText='jpg' or
                                 $SubqText='jfif' or $SubqText='png' or
                                 $SubqText='svg' or $SubqText='tif' or
                                 $SubqText='tiff' or $SubqText='ras' or
                                 $SubqText='cmx' or $SubqText='ico' or
                                 $SubqText='pnm' or $SubqText='pbm' or
                                 $SubqText='pgm' or $SubqText='ppm' or
                                 $SubqText='rgb' or $SubqText='xbm' or
                                 $SubqText='xpm' or $SubqText='xwd')">
                                    <xsl:element name="img"><xsl:attribute name="src"
                                    ><xsl:value-of select="marc:subfield[@code='u']"
                                    /></xsl:attribute><xsl:attribute name="alt"
                                    ><xsl:value-of select="marc:subfield[@code='y']"
                                    /></xsl:attribute><xsl:attribute name="style">
                                    height:100px;
                                    </xsl:attribute></xsl:element><xsl:text></xsl:tex
                                    t>
                                </xsl:when>
                                <xsl:when test="marc:subfield[@code='y' or @code='3'
                                or @code='z']">
                                    <xsl:call-template name="subfieldSelect">
                                    <xsl:with-param name="codes">y3z</xsl:with-param>
                                    </xsl:call-template>
                                </xsl:when>
                                <xsl:when test="not(marc:subfield[@code='y']) and
                                not(marc:subfield[@code='3']) and
                                not(marc:subfield[@code='z'])">
                <xsl:choose>
                <xsl:when test="$URLLinkText!=''">
                    <xsl:value-of select="$URLLinkText"/>
                </xsl:when>
```

```
 <xsl:otherwise>
     <xsl:text>Click here to access online</xsl:text>
 </xsl:otherwise>
 </xsl:choose>
                 </xsl:when>
                 </xsl:choose>
                 </a></li>
           </xsl:if>
         <xsl:if test="$OPACURLOpenInNewWindow='1'">
                <li><a target='_blank'>
<xsl:choose>
  <xsl:when test="$OPACTrackClicks='track'">
   <xsl:attribute name="href">/cgi-bin/koha/tracklinks.pl?uri=
   <xsl:value-of select="str:encode-uri(marc:subfield[@code='u'],
   true())"/>&amp;biblionumber=<xsl:value-of select="$biblionumber"/>
   </xsl:attribute>
  </xsl:when>
  <xsl:when test="$OPACTrackClicks='anonymous'">
   <xsl:attribute name="href">/cgi-bin/koha/tracklinks.pl?uri=
   <xsl:value-of select="str:encode-uri(marc:subfield[@code='u'],
   true())"/>&amp;biblionumber=<xsl:value-of select="$biblionumber"/>
   </xsl:attribute>
  </xsl:when>
  <xsl:otherwise>
            <xsl:attribute name="href"><xsl:value-of select=
            "marc:subfield[@code='u']"/></xsl:attribute>
  </xsl:otherwise>
</xsl:choose>
                 <xsl:choose>
                 <xsl:when test="($Show856uAsImage='Results' or
                 $Show856uAsImage='Both') and ($SubqText='img' or
                 $SubqText='bmp' or $SubqText='cod' or
                 $SubqText='gif' or $SubqText='ief' or
                 $SubqText='jpe' or $SubqText='jpeg' or
                 $SubqText='jpg' or $SubqText='jfif' or
                 $SubqText='png' or $SubqText='svg' or
                 $SubqText='tif' or $SubqText='tiff' or
                 $SubqText='ras' or $SubqText='cmx' or
                 $SubqText='ico' or $SubqText='pnm' or
                 $SubqText='pbm' or $SubqText='pgm' or
                 $SubqText='ppm' or $SubqText='rgb' or
                 $SubqText='xbm' or $SubqText='xpm' or
                 $SubqText='xwd')">
                     <xsl:element name="img"><xsl:attribute name="src"
                     ><xsl:value-of select="marc:subfield[@code='u']"
                     /></xsl:attribute><xsl:attribute name="alt"
                     ><xsl:value-of select="marc:subfield[@code='y']"
                     /></xsl:attribute><xsl:attribute name="style">
                     height:100px
                     </xsl:attribute></xsl:element><xsl:text></xsl:text>
                 </xsl:when>
                 <xsl:when test="marc:subfield[@code='y' or @code='3'
                 or @code='z']">
                     <xsl:call-template name="subfieldSelect">
                     <xsl:with-param name="codes">y3z</xsl:with-param>
                     </xsl:call-template>
                 </xsl:when>
                 <xsl:when test="not(marc:subfield[@code='y']) and
                 not(marc:subfield[@code='3']) and
                 not(marc:subfield[@code='z'])">
 <xsl:choose>
 <xsl:when test="$URLLinkText!=''">
     <xsl:value-of select="$URLLinkText"/>
 </xsl:when>
 <xsl:otherwise>
     <xsl:text>Click here to access online</xsl:text>
```

```
                    </xsl:otherwise>
                    </xsl:choose>
                                    </xsl:when>
                                    </xsl:choose>
                                    </a></li>
                              </xsl:if>
                            </xsl:for-each>
                            </ul>
                            </div>
                        </xsl:if>


            <!-- Local accession number -->
<!-- Modified for authcat: changed tag -->
    <xsl:if test="marc:datafield[@tag=098]">
         <div class="results_summary ">
            <span class="label">Local accession number: </span>
            <ul class="resource_list">
                <xsl:for-each select="marc:datafield[@tag=098]">
                    <li>
                        <xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">a</xsl:with-param>
                        </xsl:call-template>
                    </li>
                </xsl:for-each>
            </ul>
        </div>
    </xsl:if>



        <!-- Availability line -->
<!-- Modified for authcat: availability removed -->

        <!-- End of Availability line -->

    </xsl:template>

    <xsl:template name="nameABCQ">
            <xsl:call-template name="chopPunctuation">
                <xsl:with-param name="chopString">
                    <xsl:call-template name="subfieldSelect">
                        <xsl:with-param name="codes">abcq</xsl:with-param>
                    </xsl:call-template>
                </xsl:with-param>
                <xsl:with-param name="punctuation">
                    <xsl:text>:,;/ </xsl:text>
                </xsl:with-param>
            </xsl:call-template>
    </xsl:template>

    <xsl:template name="nameABCDN">
            <xsl:call-template name="chopPunctuation">
                <xsl:with-param name="chopString">
                    <xsl:call-template name="subfieldSelect">
                        <xsl:with-param name="codes">abcdn</xsl:with-param>
                    </xsl:call-template>
                </xsl:with-param>
                <xsl:with-param name="punctuation">
                    <xsl:text>:,;/ </xsl:text>
                </xsl:with-param>
            </xsl:call-template>
    </xsl:template>

    <xsl:template name="nameACDEQ">
            <xsl:call-template name="subfieldSelect">
                <xsl:with-param name="codes">acdeq</xsl:with-param>
            </xsl:call-template>
    </xsl:template>

    <xsl:template name="nameDate">
        <xsl:for-each select="marc:subfield[@code='d']">
            <xsl:call-template name="chopPunctuation">
                <xsl:with-param name="chopString" select="."/>
            </xsl:call-template>
        </xsl:for-each>
    </xsl:template>

    <xsl:template name="role">
        <xsl:for-each select="marc:subfield[@code='e']">
                    <xsl:value-of select="."/>
        </xsl:for-each>
        <xsl:for-each select="marc:subfield[@code='4']">
                    <xsl:value-of select="."/>
        </xsl:for-each>
    </xsl:template>

    <xsl:template name="specialSubfieldSelect">
        <xsl:param name="anyCodes"/>
        <xsl:param name="axis"/>
        <xsl:param name="beforeCodes"/>
        <xsl:param name="afterCodes"/>
        <xsl:variable name="str">
            <xsl:for-each select="marc:subfield">
                <xsl:if test="contains($anyCodes, @code) or
                (contains($beforeCodes,@code) and
                following-sibling::marc:subfield[@code=$axis]) or
                (contains($afterCodes,@code) and
                preceding-sibling::marc:subfield[@code=$axis])">
                    <xsl:value-of select="text()"/>
                    <xsl:text> </xsl:text>
                </xsl:if>
            </xsl:for-each>
        </xsl:variable>
        <xsl:value-of select="substring($str,1,string-length($str)-1)"/>
    </xsl:template>

    <xsl:template name="subtitle">
        <xsl:if test="marc:subfield[@code='b']">
                <xsl:call-template name="chopPunctuation">
                    <xsl:with-param name="chopString">
                        <xsl:value-of select="marc:subfield[@code='b']"/>

                        <!--<xsl:call-template name="subfieldSelect">
                            <xsl:with-param name="codes">b</xsl:with-param>
                        </xsl:call-template>-->
                    </xsl:with-param>
                </xsl:call-template>
        </xsl:if>
    </xsl:template>

    <xsl:template name="chopBrackets">
        <xsl:param name="chopString"></xsl:param>
        <xsl:variable name="string">
            <xsl:call-template name="chopPunctuation">
                <xsl:with-param name="chopString" select="$chopString"></xsl:with-param>
            </xsl:call-template>
        </xsl:variable>
        <xsl:if test="substring($string, 1,1)='['">
            <xsl:value-of select="substring($string,2, string-length($string)-2)"
            ></xsl:value-of>
        </xsl:if>
        <xsl:if test="substring($string, 1,1)!='['">
            <xsl:value-of select="$string"></xsl:value-of>
        </xsl:if>
    </xsl:template>

    <xsl:template name="listCallNumbers">
        <xsl:param name="items"/>
        <xsl:param name="max"/>
        <xsl:param name="status_text"/>
        <xsl:param name="class_block"/>
        <xsl:param name="class_status"/>
        <xsl:param name="OPACItemLocation"/>
        <xsl:if test="count($items)>0">
            <span><xsl:attribute name="class"><xsl:value-of select="$class_block"/>
            </xsl:attribute>
                <span>
                    <xsl:attribute name="class"><xsl:value-of select="$class_status"/>
                    </xsl:attribute>
                    <xsl:value-of select="$status_text"/>
                </span>
                <xsl:text> (</xsl:text>
                <xsl:value-of select="count($items)"/>
                <xsl:text>)</xsl:text>
                <xsl:if test="$max>0 and count($items[items:itemcallnumber!=''])>0 and
                $OPACItemLocation!='library'">
                    <span class="CallNumberAndLabel">
                    <span class="LabelCallNumber">
                        <xsl:if test="$OPACItemLocation='callnum'">Call number: </xsl:if>
                        <xsl:if test="$OPACItemLocation='ccode'">Collection, call
                        number: </xsl:if>
                        <xsl:if test="$OPACItemLocation='location'">Location, call
                        number: </xsl:if>
                    </span>
                    <span class="CallNumber">
                    <xsl:for-each select="$items[items:itemcallnumber!=''][position()
                    &lt;= $max]">
                        <xsl:if test="$OPACItemLocation='location'">
                            <strong><xsl:value-of select="concat(items:location,' ')"/>
                            </strong>
                        </xsl:if>
                        <xsl:if test="$OPACItemLocation='ccode'">
                            <strong><xsl:value-of select="concat(items:ccode,' ')"/>
                            </strong>
                        </xsl:if>
                        <xsl:value-of select="items:itemcallnumber"/>
                        <xsl:if test="position()!=last()">
                            <xsl:text>, </xsl:text>
                        </xsl:if>
                        <xsl:if test="position()=last() and (count($items)>$max or
                        count($items[items:itemcallnumber=''])>0)">
                            <xsl:text>, ..</xsl:text>
                        </xsl:if>
                    </xsl:for-each>
                    </span>
                    </span>
                </xsl:if>
                <xsl:text>. </xsl:text>
            </span>
        </xsl:if>
    </xsl:template>


</xsl:stylesheet>
```