



Managing Dates in Web Content Publisher Resources

By David Wendt (wendt@us.ibm.com)
Software Engineer, IBM Corp.
June 2002

Abstract

Web Content Publisher provides default authoring forms for add, edit, show, and preview of structured content. These forms are produced through introspection of the resource class and creation of HTML Form elements for its property types. When users (content contributors) submit the form, the data is sent as text to Web Content Publisher, which uses Java to convert the text strings to the property types of the resource. Although most Java core types are handled automatically by the generated forms, date classes are not. This paper describes how you can write custom authoring templates to handle date classes for your structured content.

Resources

Web Content Publisher manages structured content as resources usable in a WebSphere® Personalization environment. Resources generally map elements of a datastore like a database table where the properties correspond to the column types and a resource instance usually represents a single row of data from the table or view. Each column or field of data from the row is retrieved using the accessor method on the resource for the appropriate property.

Default forms

Web Content Publisher provides default authoring forms for add, edit, show, and preview. These forms are produced through introspection of the resource class and creation of HTML Form elements for its property types. When users (content contributors) submit the form, the data is sent to Web Content Publisher as text. The server code uses Java to convert these text strings to the property types. The supported property types for conversion from text are as follows:

```
java.lang.Byte  
java.lang.Integer  
java.lang.Boolean  
java.lang.Short  
java.lang.Float  
java.lang.Long  
java.lang.Double  
java.math.BigDecimal  
java.math.BigInteger
```

and primitive types:

```
byte, int, boolean, short, long, float, and double
```

The conversion also supports three date types:

```
java.sql.Time
```

```
java.sql.Date
java.sql.Timestamp
```

Unfortunately, the date types are not supported in the default authoring forms at this time. Allowing users to enter a date into an authoring form will require a custom author template.

Custom forms

Author templates allow you to substitute your own forms for the default ones presented for a content type by Web Content Publisher. These forms can be written as JSP™ or XSL and must be added as files to the Web Content Publisher project in which they will be used. The author form association is made using the Templates dialog. Wizards are available on this window to give you starter forms, which are similar to the default forms.

An author form can show any information you want in any way you want. Web Content Publisher only requires the data for the resource to be passed such that the property name identifies the property's value. This is normally through an HTML form element like the following:

```
<INPUT type="text" name="propertyName" value="">
```

This element will show a simple entry field in the browser where the user can enter data. When the form is submitted, the data entered will be passed to the authoring server and identified by the "propertyName" specified. These names are case-sensitive.

In the same way, you can specify a date property using an HTML form element. The only requirement is that you must ensure the user enters the date in the form required by the `valueOf` method for the date classes:

<code>java.sql.Time</code>	<code>hh:mm:ss</code>
<code>java.sql.Date</code>	<code>yyyy-mm-dd</code>
<code>java.sql.Timestamp</code>	<code>yyyy-mm-dd hh:mm:ss.fffffff</code>

Where `hh` is hours (0-23)
`mm` is minutes
`ss` is seconds
`yyyy` is the year
`mm` is the month (1-12)
`dd` is the day
`fffffff` is subseconds (optional)

Note that this format is locale independent. It must be passed in the above format regardless of country or language code.

You will most likely want to provide a more user-friendly interface for entering the date value. In this case, you may need to include the date property as a hidden form element as follows:

```
<INPUT type="hidden" name="myDateProperty" value="">
```

You could then use a bit of JavaScript™ to put the properly formatted date into this field before it is submitted.

The following is an example of JSP code that displays the current date elements (day, month, year) in separate combo-boxes.

```
<input type="hidden" name="myDateProperty" value="">

<table border=0 cellpadding=3 cellspacing=0>
<tr>
  <td nowrap colspan=4 align="left">Date</td>
</tr>
<tr><%
  Calendar rightNow = Calendar.getInstance();
  DateFormatSymbols symbols =
    new DateFormatSymbols( request.getLocale() );
  String[] months = symbols.getMonths();

  int currentMonth = rightNow.get(Calendar.MONTH);
  int maxDay = rightNow.getActualMaximum(Calendar.DAY_OF_MONTH);
  int currentDay = rightNow.get(Calendar.DAY_OF_MONTH);
  int currentYear = rightNow.get(Calendar.YEAR);

  %>
  <td align="left"><select name="month"><%
    for( int i=0; i<(months.length-1); i++ )
    {
      String selected = "";
      if( currentMonth==i )
        selected = "Selected";
      %>
      <option <%=selected%> value="<%= (i+1)%>">
        <%= months[i] %></option><%
    }
    %>
  </select>
</td>
  <td align="left"><select name="day"><%
    for( int i=0; i<maxDay; i++ )
    {
      String selected = "";
      if( currentDay==(i+1) )
        selected = "Selected";
      %>
      <option <%=selected%> value="<%= (i+1)%>"><%= (i+1)%></option><%
    }
    %>
  </select>
</td>
  <td align="left"><select name="year">
    <option selected value="<%=currentYear%>">
      <%=currentYear%></option>
    <option value="<%= (currentYear+1)%>">
      <%= (currentYear+1)%></option>
```

```

        <option value="<%= (currentYear+2)%>">
            <%= (currentYear+2)%></option>
    </select>
</td>
</tr>
</table>

<input type="button" value="Add" onclick="submitForm()">
<script language="JavaScript">
    function submitForm()
    {
        var month = document.forms[0].month.value;
        var day   = document.forms[0].day.value;
        var year  = document.forms[0].year.value;

        // validate values for date
        // and other fields in the form

        // format the date for java.sql.Date.valueOf
        document.forms[0].myDateProperty.value = year + "-" + month +
                                                    "-" + day;

        // submit the form
        document.forms[0].submit();
    }
</script>

```

Users select the date values using the combo-boxes and click the “Add” button. The button invokes a JavaScript method, which validates the form values and, in this case, formats the date value for the hidden myDateProperty field appropriately. Web Content Publisher will know to convert the value for myDateProperty based on the property-type (in this case `java.sql.Date`) by passing the formatted string to the `valueOf` method or `java.sql.Date`.

For more information on developing custom templates, refer to the Web Content Publisher Help System.

Trademarks

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and other countries.

ActiveX, Microsoft, Windows, Windows NT[®], and the Windows logo are trademarks or registered trademarks of Microsoft Corporation in the United States, or other countries, or both.

UNIX is a registered trademark of The Open Group.

Linux is a registered trademark of Linus Torvalds.

Other company, product, and service names, which may be denoted by a double asterisk(**), may be trademarks or service marks of others.

Notices

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law:

INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION AND ANY ASSOCIATED CODE "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

© Copyright International Business Machines Corporation 2002. All rights reserved. US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.