



# Getting Resources in Authoring Templates in Web Content Publisher

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. Customers can substitute their own authoring forms to address any content type. This paper describes how to access structured content from custom authoring forms and JSP™ generation templates.

## Structured content

Web Content Publisher manages structured content as resource collections. Structured content is stored in a format that preserves not just the data but information about the data. A resource is a Java™ class that defines the properties of a content object.

Two Java classes are provided for accessing the resources in a project:

```
com.ibm.wcm.GetGenericResource (get a single resource)
com.ibm.wcm.GetGenericResourceList (get a list of resources)
```

These classes are used in authoring and generation templates for resolving resources appropriately in a JSP. The sample authoring forms and generation templates as described in the *Web Content Publisher Guide* show the use of these two classes via the `<jsp:useBean>` tag. They can also be used directly in Java code inside a JSP to retrieve resources.

## GetGenericResource

You can invoke the edit, show, and preview authoring templates for a particular resource using the following URL format:

```
.../authoringEdit.jsp?beanName=WCMSample.Product&id=FT0100
```

Where `WCMSample.Product` is the class name of the resource and the `FT0100` is the resource ID intended for use in the template.

The following `useBean` tag in the authoring template JSP file will retrieve the resource object using the request parameters and assign it a request-scoped attribute named `resObj`.

```
<jsp:useBean class="com.ibm.wcm.GetGenericResource"
             id="getResource"
             type="com.ibm.wcm.GetGenericResource">
<% getResource.getResource(request); %>
</jsp:useBean>
```

The HTTP request parameters passed to the JSP file and expected by the `useBean` tag:

```
beanName = class name of the resource type
id       = resource Id of the resource to retrieve
```

The following `useBean` dereferences the request-scoped attribute from the request so that it may be used as a typed variable called `resObj` in the JSP logic.

```
<jsp:useBean id="resObj" type="WCMSample.Product" scope="request"/>
```

You cannot change the name of the `resObj` variable.

You can also use `GetGenericResource` by instantiating a new one (using the new operator) or by reusing the `resObj` reference created by the `useBean` tags as described above.

The following is an example showing a new instance of `GetGenericResource` retrieving various resources of different types.

```
com.ibm.wcm.GetGenericResource getResource =
    new com.ibm.wcm.GetGenericResource();
getResource.setRequest(request);

WCMSample.Product product1 =
    (WCMSample.Product)getResource( "WCMSample.Product", "FT0100" );

WCMSample.Product product2 =
    (WCMSample.Product)getResource( "WCMSample.Product", "FT0200" );

WCMSample.CompanyNews news1 =
    (WCMSample.CompanyNews)getResource( "WCMSample.CompanyNews",
                                         "IBM Stock Soars!" );
```

The first two lines initialize a `GetGenericResource` object with the `request` object. The `request` object is very important because it provides Web Content Publisher with the project information context under which the resources should be retrieved. This `GetGenericResource` object can be used to retrieve as many resources of any type as required from the current project for use in the JSP.

Note that `GetGenericResource` cannot be used outside of the authoring environment and will not work in a JSP published to a run-time environment where Web Content Publisher is not used to invoke the JSP. This is because an authoring-environment request object is required in the `setRequest()` method in order to retrieve content.

### **GetGenericResourceList**

The `GetGenericResourceList` class is intended for use in a summary generation JSP template but can also be used in authoring forms where a list of resources may be needed for reference in a multivalue field.

The summary template is invoked passing HTTP request parameters using the following URL format:

```
.../summary1.jsp?beanName=WCMSample.Product&path=Dolls/  
-or-  
.../summary1.jsp?beanName=WCMSample.Product&path=CheapToys&view=1
```

Where `WCMSample.Product` is the class name of the resource, `Dolls` is a folder in the project, and `CheapToys` is a search view within the resource collection.

The following `useBean` tag is used to retrieve the request parameters and then get the list of resources for the path specified.

```
<jsp:useBean class="com.ibm.wcm.GetGenericResourceList"  
            id="getResourceList"  
            type="com.ibm.wcm.GetGenericResourceList">  
<% getResourceList.setRequest(request); %>  
</jsp:useBean>
```

The HTTP request parameters for the JSP are expected by this `useBean` tag:

```
beanName = class name of the resource type  
id       = resource Id of the resource to retrieve
```

The list is retrieved by getting the enumeration of the resource list.

```
<%  
    java.util.Enumeration enumObjs = getResourceList.getResourceList();  
%>
```

`GetGenericResourceList` can also be reused repeatedly to get other lists of resources. However, arbitrary queries are not currently supported. You must specify either a folder name or a view name for the current project. The following `GetGenericResourceList` methods are available:

```
public Enumeration getResourceListFromPath( String beanName,  
                                           String path )  
public Enumeration getResourceListFromView( String beanName,  
                                           String view )
```

They can be used directly as follows:

```
com.ibm.wcm.GetGenericResourceList getResourceList =  
    new com.ibm.wcm.GetGenericResourceList();  
getResourceList.setRequest(request);  
  
Enumeration enumDolls =  
    getResourceList.getResourceListFromPath( "WCMSample.Product",  
                                           "Dolls/" );  
  
Enumeration enumScienceNews =
```

```
getResourceList.getResourceListFromPath( "WCMSample.CompanyNews",  
                                         "Science/Space/" );  
  
Enumeration enumCheapToys =  
    getResourceList.getResourceListFromView( "WCMSample.Product",  
                                             "CheapToys" );
```

The first two lines initialize a `GetGenericResourceList` object with the request object. The `request` object is very important since it provides the project information context to Web Content Publisher under which the resources should be retrieved. This `GetGenericResourceList` object can be used to retrieve as many resources of any type as required from the current project for use in the JSP.

Note that `GetGenericResourceList` cannot be used outside of the authoring environment and will not work in a JSP published to a run-time environment where Web Content Publisher is not used to invoke the JSP. This is because an authoring-environment request object is required in the `setRequest()` method in order to retrieve content.

For more information on using 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<sup>®</sup>, 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.