

FatWire | Content Server 7

Version 7.5

Internationalization Settings Guide

Document Revision Date: Jun. 15, 2011



FATWIRE CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. In no event shall FatWire be liable for any loss of profits, loss of business, loss of use of data, interruption of business, or for indirect, special, incidental, or consequential damages of any kind, even if FatWire has been advised of the possibility of such damages arising from this publication. FatWire may revise this publication from time to time without notice. Some states or jurisdictions do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

Copyright © 2008–2011 FatWire Corporation. All rights reserved.

This product may be covered under one or more of the following U.S. patents: 4477698, 4540855, 4720853, 4742538, 4742539, 4782510, 4797911, 4894857, 5070525, RE36416, 5309505, 5511112, 5581602, 5594791, 5675637, 5708780, 5715314, 5724424, 5812776, 5828731, 5909492, 5924090, 5963635, 6012071, 6049785, 6055522, 6118763, 6195649, 6199051, 6205437, 6212634, 6279112 and 6314089. Additional patents pending.

FatWire, Content Server, Content Server Bridge Enterprise, Content Server Bridge XML, Content Server COM Interfaces, Content Server Desktop, Content Server Direct, Content Server Direct Advantage, Content Server DocLink, Content Server Engage, Content Server InSite Editor, Content Server Satellite, and Transact are trademarks or registered trademarks of FatWire Corporation in the United States and other countries.

iPlanet, Java, J2EE, Solaris, Sun, and other Sun products referenced herein are trademarks or registered trademarks of Sun Microsystems, Inc. *AIX, IBM, WebSphere*, and other IBM products referenced herein are trademarks or registered trademarks of IBM Corporation. *WebLogic* is a registered trademark of BEA Systems, Inc. *Microsoft, Windows* and other Microsoft products referenced herein are trademarks or registered trademarks of Microsoft Corporation. *UNIX* is a registered trademark of The Open Group. Any other trademarks and product names used herein may be the trademarks of their respective owners.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>) and software developed by Sun Microsystems, Inc. This product contains encryption technology from Phaos Technology Corporation. This product includes software developed by DOM4J Project (<http://www.dom4j.org>). This product includes software developed by the Indiana University Extreme! Lab. For further information please visit <http://www.extreme.indiana.edu/>.

You may not download or otherwise export or reexport this Program, its Documentation, or any underlying information or technology except in full compliance with all United States and other applicable laws and regulations, including without limitations the United States Export Administration Act, the Trading with the Enemy Act, the International Emergency Economic Powers Act and any regulations thereunder. Any transfer of technical data outside the United States by any means, including the Internet, is an export control requirement under U.S. law. In particular, but without limitation, none of the Program, its Documentation, or underlying information of technology may be downloaded or otherwise exported or reexported (i) into (or to a national or resident, wherever located, of) Cuba, Libya, North Korea, Iran, Iraq, Sudan, Syria, or any other country to which the U.S. prohibits exports of goods or technical data; or (ii) to anyone on the U.S. Treasury Department's Specially Designated Nationals List or the Table of Denial Orders issued by the Department of Commerce. By downloading or using the Program or its Documentation, you are agreeing to the foregoing and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list or table. In addition, if the Program or Documentation is identified as Domestic Only or Not-for-Export (for example, on the box, media, in the installation process, during the download process, or in the Documentation), then except for export to Canada for use in Canada by Canadian citizens, the Program, Documentation, and any underlying information or technology may not be exported outside the United States or to any foreign entity or "foreign person" as defined by U.S. Government regulations, including without limitation, anyone who is not a citizen, national, or lawful permanent resident of the United States. By using this Program and Documentation, you are agreeing to the foregoing and you are representing and warranting that you are not a "foreign person" or under the control of a "foreign person."

Internationalization Settings Guide

Document Revision Date: Jun. 15, 2011

Product Version: 7.5

FatWire Technical Support

www.fatwire.com/Support

FatWire Headquarters

FatWire Corporation
330 Old Country Road
Suite 303
Mineola, NY 11501
www.fatwire.com

Table of Contents

About This Guide	5
Who Should Use This Guide	5
Related Publications	5
1 Configuring Your Environment	7
CS-Specific Settings	8
cs.contenttype Property	8
cs.contenttype Variable (Set in SiteCatalog resargs1)	8
_charset_hidden Variable in Forms	8
Preferred Encoding	9
Encoding Specified in XML or JSP Elements	9
Using SetVar Tag	9
Using HTTP META tag	10
Internet Explorer Settings	10
2 Areas to Watch Out For	11
Files Stored on the File System	12
XML and JSP Files	12
HTML Files	12
SystemSQL Queries	12
Page Cache Files Referenced from the SystemPageCache table	12
Attribute Editor	12
Article Bodies, Flex Assets, User-Defined Assets	12
eWebEditPro	12
XML Post	13
Catalog Mover	13
Content Server Explorer	13
CS-Desktop and CS-DocLink	13
Content Server's Interfaces	14
Single-Language Restrictions	14
Functional Restrictions	15

About This Guide

This guide provides instructions for configuring your environment to support the UTF-8 character encoding used by Content Server.

Who Should Use This Guide

This guide is written for installation engineers, developers, and administrators of Content Server systems. Users of this guide must be familiar with databases, application servers, web servers, and browsers; with file system encodings and product-specific property files and tags. For CS-based operations, users of this guide must also be familiar with Content Server's interfaces, such as Advanced, Dash, and InSite, as well as clients, such as CS-Desktop and CS-DocLink.

Related Publications

The FatWire library includes publications written for Content Server developers and administrators. The publications are provided as product manuals with your Content Server installation. They are also posted on the Web at the following URL:

<http://e-docs.fatwire.com>

The documentation website is password-protected. You can obtain a password from FatWire Technical Support (click **Request for a New Account**). More information about FatWire Technical Support is available at the following website:

<http://www.fatwire.com/Support>

Other publications, such as white papers, provide information about Content Server's feature set and its business applications. To obtain these publications, contact sales@fatwire.com.

Chapter 1

Configuring Your Environment

This chapter describes how to configure your environment to enable multi-language operations in Content Server.

This chapter contains the following sections:

- [CS-Specific Settings](#)
- [Internet Explorer Settings](#)

CS-Specific Settings

Note

- The settings that you choose for a given Content Server instance must be reproduced on all cluster members (if any) and across all environments (development, management, delivery, and so on).
- If none of the properties and variables described below are set, the `cs.contentType` property defaults to `text/html`. The character set of the output now defaults to default system encoding.

cs.contentType Property

`cs.contentType` is a system-wide (global) property that defines the outgoing character encoding. By default, the property is set to `text/html; charset=UTF-8`. If you need a specific encoding, change the value. For example, if you want the outgoing encoding to be Shift JIS, set this property to `text/html; charset=Shift_JIS`. Content Server Explorer depends on this setting to display data correctly.

cs.contentType Variable (Set in SiteCatalog resargs1)

The `cs.contentType` variable enables you to control the outgoing encoding on a page-by-page basis. The value should be set in the same way as the `cs.contentType` property, as shown in “[cs.contentType Property](#).” This variable overrides the value defined in `cs.contentType` property.

Note that pages under WebServices are set to `cs.contentType=application/xml; charset=UTF-8`.

_charset_hidden Variable in Forms

If you are using forms to input international data, make sure that you set this input type variable at the very beginning, after the form declaration. For example, if you expect the user to use non-ASCII data, make sure that the form looks similar to the following:

```
<form action='ContentServer' method='get'>
<input type='hidden' name='_charset_'/>
<input type='hidden' name='pagename'
  value='<%=ics.GetVar("pagename")%>' />
<input type='text' name='name' value='<%=ics.GetVar("name")%>' />
<input type='submit' />
</form>
```

One point to note is that the `_charset_hidden` variable without a value works only on the Internet Explorer browser.

Preferred Encoding

When Content Server needs to consume HTTP requests with certain encodings (Cp943C for example) that are closely related to a more widely used encoding (Shift_JIS), it is not sufficient to merely rely on the `_charset_` hidden variable alone. Internet Explorer, when it encounters a `_charset_` value set to Cp943C, changes it to Shift_JIS. This forces Content Server to read all data in Shift_JIS. To overcome this, a special names property syntax is used:

```
cs.contentType.<charset>=<preferred_encoding_for_this_charset>
```

For example, in relation to the scenario described above, this property would be specified as follows to indicate to Content Server to use Cp943C:

```
cs.contentType.Shift_JIS=Cp943C
```

Note that this property structure is necessary only in special circumstances such as the one described above, where the behavior of Internet Explorer conflicts with and changes the value of `_charset_` value.

Encoding Specified in XML or JSP Elements

The encoding in the `<?xml` line of an XML element specifies the encoding of the `.xml` file on disk. The same is true of JSP. The encoding specified in the page directive specifies two things. The first is the encoding of the `.jsp` file on disk. The second is the outgoing encoding of the evaluated JSP element. This gets converted to the encoding of the enclosing JSP, or in the XML case, the outgoing encoding of the page (content-type). So `cs.contentType` can be used to indicate that the outgoing page will have a specific encoding, like Shift-JIS, but a JSP can output UTF-8, and the UTF-8 will get converted to Shift-JIS into the output stream of the page response. An example on how to specify the encoding is:

- XML: `<?xml version="1.0" encoding="UTF-8"?>`
- JSP: `<%@ page contentType="text/html; charset=UTF-8" %>`

Using SetVar Tag

You can also control the outgoing page encoding by using the `SetVar` tag in JSP and XMLs. The catch in using the tag is that it must be set before anything is streamed out.

In JSPs we can do:

```
<cs:ftcs>
<ics:setvar name="cs.contentType" value="text/html; charset=UTF-8"
/>
...
</cs:ftcs>
```

In XML, you have the following options:

```
<ftcs>
<setvar name="cs.contentType" value="text/html; charset=utf-8"/>
...
</ftcs>
```

The second option is to use the `ics.streamheader` XML tag, but again this must be done before anything is streamed, and only in XML.

```
<ftcs>
<ics.streamheader name="Content-Type" value="text/html;
  charset=utf-8"/>
...
</ftcs>
```

Using HTTP META tag

If the encoding is specified by any of the methods above, then the META tag has no effect. Otherwise, the browser tries to display the data in the encoding specified by the META tag.

Internet Explorer Settings

IE 6 and above by default have all the languages installed. If you are unable to see text in a particular language, you most probably need to enable it in your browser.

To view content in different languages

1. Go to **Tools > Internet Options > General** tab.
2. Click on **Languages** on the bottom right of the page.
3. Click **Add** to add more languages to the list of languages already displayed in the box.
4. Click **OK**.
5. Close and reopen Internet Explorer. You should now see the content in your specified language if the web page so provides.

Chapter 2

Areas to Watch Out For

The following section lists several features in Content Server with specific internationalization requirements. The features are:

- [Files Stored on the File System](#)
- [Attribute Editor](#)
- [eWebEditPro](#)
- [XML Post](#)
- [Catalog Mover](#)
- [Content Server Explorer](#)
- [CS-Desktop and CS-DocLink](#)
- [Content Server's Interfaces](#)

Files Stored on the File System

Many indirect files are stored on the file system because of the `url` column references. The files are identified in this section.

XML and JSP Files

See “[Encoding Specified in XML or JSP Elements](#),” on page 9.

HTML Files

The HTML files are read using the `file.encoding` Java parameter value. The data in the file also depends on the way it was stored initially.

SystemSQL Queries

SystemSQL uses a `url` column to point to a file on the file system that holds a SQL query. When this file is loaded, it is assumed that the encoding of the file is the Java default encoding (`System.getProperty("file.encoding")`). There are several possible ways to create the SystemSQL queries (CSE, text editor, Content Server application). It is probably best to always stick to ASCII7 when creating queries, since any data is typically merged using variable replacement at run time.

Page Cache Files Referenced from the SystemPageCache table

For page cache files, we manage the page cache file so that the encoding of the file is UTF-8. Since we write and read this file only through Content Server, it can be managed this way.

Attribute Editor

You have two ways to specify the text for an attribute editor:

- Type in the text in the text area provided. The form post will determine the encoding.
- Use the **Browse** button to select a text file. The text file encoding should match the encoding specified in `xcelerate.charset` encoding.

Article Bodies, Flex Assets, User-Defined Assets

The article bodies are stored on the file system using the `file.encoding` Java property value.

eWebEditPro

To manually set eWebEditPro to use UTF-8, change the value for `charencode` from `charref` to `utf-8` in the file `eWebEditPro/config.xml`. For details, refer to this URL: http://www.ektron.com//support/ewebeditprokb.cfm?doc_id=1229

XML Post

When posting non ASCII files using XML Post, the java file encoding must match the encoding of the file. For example, if you are posting a Japanese file (stored as UTF-8) to a UTF-8 system, then one of the following should be set before the XML Post command is run:

- The system locale must be set to UTF-8.
- The option of `-Dfile.encoding=UTF-8` must be specified in the XML Post command.

Similarly, if the file is stored as Shift_JIS, then the corresponding system locale should be set or the `java file.encoding` option must be specified.

Content Server supports the encoding in the `<?xml` line, as the first line in the posted XML file. This overrides everything else as far as the encoding in which the `.xml` file is read.

Catalog Mover

Edit the catalog mover `.bat` (or `.sh` on UNIX) file and modify the java command to include the `file.encoding` parameter with a value that reflects the correct encoding needed to display the characters stored in the catalogs. This step can be avoided if the default encoding of the file system matches that of the data stored in the catalogs.

Content Server Explorer

Content Server Explorer requires that `cs.contenttype` property in `futuretense.ini` be set to the correct value. If you are viewing simple ASCII characters, nothing needs to be done. However, for viewing complex characters, such as Japanese, you need to set `cs.contenttype` to one of the following:

- `text/html; charset=SJIS`
- `text/html; charset=UTF-8`

Also, users may need to load font support for Japanese and various other character sets in order to have them display correctly. To do this in Windows 2000 for example, go to **Settings > Control Panel > Regional Options**. In the first tab, **General**, select the languages you want to support from the list titled “Language settings for the system.” Click **Apply** then **OK**. At this point you will be required to put in the Windows installation CD.

CS-Desktop and CS-DocLink

The CS-Desktop and CS-DocLink clients support the character sets supported by Windows. In order to enable a specific character set in CS-Desktop or CS-DocLink, first enable it in Windows. (For instructions, see the Microsoft Windows documentation.)

Content Server's Interfaces

The user's machine must be able to support the characters that are to be displayed in Content Server's interfaces. For languages other than English, the user needs to make sure of the following:

- The appropriate fonts to display the characters are installed.
- For a Windows machine, the locale and language settings must support the characters coming through. For example, for the interfaces to be able to display Japanese characters, Windows must first be configured to display those characters. (For instructions on configuring Windows to support the target language, see the Windows documentation.)
- For a UNIX machine, the locale (`LANG` and `LC_ALL` environment variables) must be appropriately set.
- The browser's encoding must be correctly set.

Single-Language Restrictions

Although you can configure a multi-lingual management system, certain parts of the user interface can be displayed in one language only.

For example, the names of tables and columns in the Content Server database as well as individual items such as categories and source codes can have one name only. This means that although much of the text on an individual Content Server form can be displayed in multiple languages, items such as field names and asset type names can be displayed in one language only.

Following is a list of items in CS that can have one name only, which means that they can be displayed in one language only:

- Asset type names
- Field names
- Asset names
- Categories
- Source codes
- Tree tab names
- Site names
- Names of workflow building blocks (actions, e-mail objects, conditions, states, steps, processes)
- Role names
- Start menu items—both Search and New

On a system that supports two or more languages, you must determine which language is going to be used by the majority of content providers and then use that language to name your sites, tabs, asset types, and so on.

Functional Restrictions

Content Server has the following functional restrictions for international use:

- The Property Editor supports ASCII only.
- Decimal numbers must be entered in US format (“.” decimal separator).

