Space Details

Key:	PentahoDoc
Name:	BI Server Documentation - Latest
Description:	Latest version of the Pentaho BI Server
Creator (Creation Date):	admin (Nov 15, 2006)
Last Modifier (Mod. Date):	admin (Nov 27, 2006)

Available Pages

Internationalization Guide

This page last changed on Nov 29, 2006 by mbatchelor.

This guide will help you translate the Pentaho BI Platform from English to another language. You can do this using the Pre-Configured Installation and a text editor. All of the messages for the user interface and log files are stored in text files. The names of these text files all end in '.properties' and are located in many folders. To support a new language you need to create .properties files for your language. There is a standard way to name these files, and we have created the files for several requested languages.

> In constructing the Pentaho BI Platform, we have adhered to Sun's standards for internationalizing a Java application. This should make preparing the platform for any language a matter of translation and not software modification. To read about Sun's internationalization standards and how to internationalize a Java application, we recommend the following reads:
>
> - Internationalize Your Software, Jeff Friesen, Java World
> - Java Internationalization: An Overview, John Connor, Sun Microsystems Inc.
> - Java Internationalization: Localization with Resource Bundles, John Connor, Sun Microsystems Inc.
> - Internationalization, O'Rielly ONJava.com, David Flanagan
>
> This Internationalization Guide does not cover how to translate the sample data that is used by the samples. This will be covered in future editions.

Document Sections

01. Getting Started

This page last changed on Nov 29, 2006 by mbatchelor.

Internationalization Guide 02. General Concepts

1. First visit the Internationalization Forum at http://www.pentaho.org/discussion to see if other people have started translations for your language.
2. If you want to translate into French, German, Spanish, Italian, Portuguese, or Chinese we have created the .properties files for you. If you want to translate into another language email communityconnection@pentaho.orgor post onto the Pentaho forums.
3. Download the latest version of the Pre-Configured Installation which is labeled 'pentaho_demo-x.x.x.x.zip' from SourceForge. Make sure you know how to run the demo web pages, portal pages and samples. Use the 'Getting Started Guide' and the forums for help.
4. Look in the Pre-Configured Installation and 'pentaho-solutions' directories for files with names ending in .properties. You will see
    - *.properties (English)
    - *_fr.properties (French)
    - *_de.properties (German)
    - *_it.properties (Italian)
    - *_es.properties (Spanish)
    - *_pt.properties (Portugese)
    - *_cn.properties (Chinese)

If your language is not in this list, or you wish to create a country-specific version, e.g. French Canadian, contact communityconnection@pentaho.org and we will help you. For a list of the standard language and country codes read these web sites:

- Language codes: http://ftp.ics.uci.edu/pub/ietf/http/related/iso639.txt
- Country codes: http://userpage.chemie.fu-berlin.de/diverse/doc/ISO_3166.html

02. General Concepts

This page last changed on Nov 29, 2006 by mbatchelor.

01. Getting Started Internationalization Guide 03. Translating the Samples

All text displayed to the user is stored in .properties files. Whenever a piece of text is required by the platform, the appropriate .properties files are located and the required text is read from them. There are .properties files for each language that the system supports. The two character language code and optionally, the two character country code are appended to the base filename to identify the locale supported by the file. For example:

messages.properties: The default and English version of the messages file
messages_fr.properties: The French language version of the messages file
messages_de.properties: German language version of the messages file
messages_fr_CA.properties: The French Canadian version of the messages file

When a request comes in from a user, the platform detects the country and language for that user and locates the correct language version of the file. The .properties files are read at runtime and not compiled into the platform. This makes performing the translation and seeing the results a simple process.

03. Translating the Samples

This page last changed on Nov 29, 2006 by mbatchelor.

02. General Concepts | Internationalization Guide | 04. Translating the Pentaho Application

The easiest way to get started with translating the platform is by translating the samples. The 'pentaho-solutions' directory in the Pre-Configured Installation contains all of the files necessary to run the sample solutions. All of the text in the .properties files will be seen by users and should be translated. The root directory for these files is:

<pentaho-demo>/pentaho-solutions/

Each directory under the pentaho-solutions contains an **index.xml** file and a corresponding **index.properties** file. For example, when the **index.xml** file under **pentaho-solutions/samples** is requested by a user with an English locale, the platform will look for an **index.properties** file in the same directory. When someone with a French locale logs in, the platform will look for **index_fr.properties**. And for a German locale, the platform looks for an **index_de.properties**. If the **index_[country-code].properties** file is not found, the default **index.properties** is used. If you are a German translator, you will create a new file called **index_de.properties**, add the German translated text and reload your browser. The German messages will now be used for German users. In order to select the correct text for the correct messages, there is a mapping scheme between the base document and the text in the .properties file. For the example **index.xml**, here is the text in the file:

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<index>
       <name>%name</name>
       <description>%description</description>
       <icon>solutions.png</icon>
       <visible>true</visible>
</index>
```

Here are the contents of the **index.properties** file:

name=Pentaho BI Components Demo
description=These sample JSP pages show ...

For translating XML files, we use the Eclipse standard of mapping XML text beginning with a percent '%' sign to a name in the .properties file. In this example, **%name** in the XML file will be replaced by the text after **name=** key the .properties file. The index.xml files are used by the samples navigation JSP to generate names, descriptions and the image to use for a set of solutions. Here are the contents of the index_fr.properties:

name=Démo De Composants de BI De Pentaho
description=Ces pages témoin JSP montrent comment ...

Note: This example French translation was performed by a translation program and should not be taken literally. Steps for translating the samples:

1. Start with the samples directory
2. Open each .properties file for your language

3. Edit the English text on the right of the equal '=' sign with the correct text for your language. IMPORTANT: DO NOT CHANGE THE TEXT TO THE LEFT OF THE '='. e.g. name=Démo De Composants

Seeing the results of your samples translations

As you translate each file, or each line, you can view the results. Some of the text will change just by refreshing your browser. The names and descriptions of Action documents are cached and must be published. To publish, navigate to the top level of the Pentaho Preconfigured Install Demo. The default URL is http://localhost:8080/pentaho. In the Refresh section, click "Solution Repository." A pop-up dialog will load. From the list, click the "Publish" link for the "Solution Repository" publisher. It is helpful if you keep this page available while translating.

There are three file types that will be encountered while translating the samples. After translating their .properties file, the changes can be verified as follows:

1. **index.xml**-- Used for navigating between the groups of samples. You can see the results of changes to the index.properties files by navigating to that group. "Getting Started" is one such group.
2. ***.url**-- Define URLs that are displayed along with the samples and can be used to initiate actions. The SchedulerAdmin under the admin folder is an example.
3. ***.xaction**. Action documents that implement BI solutions. Select the action to verify your language changes.

When you are ready to share your work you can send it to Pentaho to be included in the Platform.

04. Translating the Pentaho Application

This page last changed on Nov 29, 2006 by mbatchelor.

03. Translating the Samples | Internationalization Guide | 05. Translating the Pentaho Report Design Wizard

The .properties files in the Pentaho application directory tree contain text used in the application files. These include error messages, debug messages, and user interface text. These files are all named 'messages.properties'. In the Pre-Configured install, the application resource bundles are located below the WEB-INF directory. By default, this is the path to the WEB-INF directory:

/pentaho-demo/jboss/server/default/deploy/pentaho.war/WEB-INF

Not all of the text in these files will be seen by users during typical use. To make this clear we have used a naming convention for each piece of text in the .properties files.

Prefix	**Meaning**
.USER_	This text will be viewable by users.
.ERROR_	This text represents an error in the system. Viewable in event logs if log level is ERROR or above.
.WARN_	This text represents a warning in the system. Viewable in event logs if log level is WARN or above.
.INFO_	This text represents an error in the system. Viewable in event logs if log level is INFO or above.
.DEBUG_	This text represents detailed system information used for problem tracking. Viewable in event logs if log level is DEBUG or above.

The priority order in which you should translate the messages is:

1. All of the text in WEB-INF\portlet.properties
2. Translate according to the following table:

For the following paths	**Translate text names with**
WEB-INF\classes\org\pentaho\locale WEB-INF\classes\org\pentaho\ui\component	.USER_ .ERROR_ .WARN_ .DEBUG_
WEB-INF\classes\org\pentaho\ui\portlet	.USER_ .ERROR_ .WARN_ .DEBUG_
WEB-INF\classes\org\pentaho\ui\component\charting	USER_

	.ERROR_ .WARN_ .DEBUG_
WEB-INF\classes\org\pentaho\ui\portlet\charting	.USER_ .ERROR_ .WARN_ .DEBUG_
WEB-INF\classes\org\pentaho\ui\servlet	.USER_ .ERROR_ .WARN_ .DEBUG_

If you just want to translate the Pre-Configured Installation and samples, you only need to translate the following files under **<pentaho-demo>/jboss/server/default/deploy/pentaho.war/WEB-INF**:

\portlet.properties
\classes\org\pentaho\locale\messages.properties

This is significantly different than prior releases of the Pentaho Pre-Configured install. Previously each Java package owned its own ResourceBundle manager 'Messages.java' and a set of properties files called messages.properties (one for each locale). Now we have consolidated all of our resources into the directories listed above using a common set of manager classes, org\pentaho\messages. There is a messages.properties for each locale and we have merged all previously translated bundles into one locale specific counterpart. This should make it much easier to locate strings that need to be translated as they all live in a common location.

Like the samples, the format of the .properties file is **name=message**. You must be careful not to change anything to the left of the equals '=' sign. All messages must stay on one line. Do NOT add carriage returns to the .properties files.

In the application messages, there are special replaceable parameter tags. They look like {0} and {1} etc. They are used to build messages. For example:

HelloWorld.USER_HELLO_WORLD_TEXT=\nHello World.{0}\n

The {0} represents a placeholder, and SHOULD NOT BE CHANGED. The text for these placeholders will come from data and be inserted by the application. All of the placeholders from the original text must appear in the translated string. The test position of the placeholders and the order they appear may be changed in order to match the correct grammar of your language. So, our second example would look like

HelloWorld.USER_HELLO_WORLD_TEXT =\nBonjour Monde.{0}\n

Seeing the results of your application translations

In order to see your changes you will have to restart the application server. Use 'stop-pentaho' and 'start-pentaho' to do this. The name of the text may help figure out where to look for your results. In this example, you would look for a message displayed to the user while executing the "Hello World" component.

Testing From Your Browser

If you want to switch between languages in your web browser you can do so by changing the language preference of your web browser.

Changing Language Preferences in Internet Explorer

In Internet Explorer, follow these steps to change your language preference:

1. From the main menu, navigate to the Tools menu, then select Internet Options...
2. On the General tab, select the Languages... button.
3. On the Languages dialog, select the Add... button.
4. You will get a dialog with a list of locales to choose from. Select the locale you wish to change your preference to, then select OK.
5. Your selection should appear in the Languages dialog now. Select your preference, then select the Move Up button until your preference is first in the list.
6. Select OK until all dialogs are closed.
7. Refresh your web page.

Changing Language Preferences in Firefox

In Mozilla Firefox, follow these steps to change your language preference:

1. From the main menu, navigate to the Tools menu, then select Options...
2. On the Advanced tab, select the Edit Languages... button.
3. On the Languages dialog, select the combo box at the bottom of the dialog.
4. You will get a list of locales to choose from. Select the locale you wish to change your preference to.
5. Select the Add button to add your preference to the displayed list.
6. Your selection should appear in the Edit Languages dialog list now. Select your preference in the list, then select the Move Up button until your preference is first in the list.
7. Select OK until all dialogs are closed.
8. Refresh your web page

05. Translating the Pentaho Report Design Wizard

This page last changed on Nov 29, 2006 by mbatchelor.

04. Translating the Pentaho Application | Internationalization Guide | 06. Translating the Design Studio

The Report Design Wizard ships with a single messages.properties which contains all of the base resource strings. If you would like to translate the Report Design Wizard you'll have to find the pentaho-reporting-X.X.X.jar in the RDW/lib directory. Extract this jar using an unzip utility into RDW/resources, meaning an "org" folder is created in RDW/resources. You may now delete the original jar file (to remove it from the classpath).

Navigate into org/pentaho/jfreereport/messages (under "lib"). Create a .properties file for your locale following the naming conventions outlined earlier. So if you have German .properties create a messages_de.properties. For each resource you want to localize, copy the key from the base messages.properties into the locale specific properties file, and provide the translation for the value (right hand side).

You may continue to run in this mode and do not need to re-bundle anything to test or deliver back to Pentaho. When you are satisfied with the localization simply provide us with your messages_XX.properties file.

06. Translating the Design Studio

This page last changed on Nov 29, 2006 by mbatchelor.

05. Translating the Pentaho Report Design Wizard Internationalization Guide 07. Unicode Support

The Pentaho Design Studio ships with a single messages.properties which contains all of the base resource strings. If you would like to translate the Design Studio you'll have to find the pentaho-action-editor.jar in the plugins\org.pentaho.workbench.editors.actionsequence_x.x.x.x directory of your Design Studio (Eclipse) installation. Extract this jar using an unzip utility into a working directory.

Navigate into org/pentaho/workbench/messages. Create a .properties file for your locale following the naming conventions outlined in the General Concepts section. So if you have German .properties create a messages_de.properties. For each resource you want to localize copy the key from the base messages.properties into the locale specific properties file and provide the translation for the value (right hand side).

Zip the contents of your working directory and naming the archive pentaho-action-editor.jar, and copy the new jar back into the plug-in directory.

07. Unicode Support

This page last changed on Nov 29, 2006 by mbatchelor.

06. Translating the Design Studio | Internationalization Guide | 08. Submitting Your Translation

Unicode is supported by the Pentaho BI Platform. Follow these guidelines for successful translations.

Properties Files

These files will be read as ISO-8859-1 encoded files. To store Unicode characters in these files they must be encoded using Java, **\uNNNN**, or XML, **NNNN;**, encoding and stored as ASCII files. You can use a text editor such as SC Unipad (http://www.unipad.org) to edit the Pentaho .properties files and save them in the correct format.

XML Files

These files must be stored with all Unicode characters encoded using XML, **NNNN;**, encoding.

XSL Files

Unicode characters stored in the XSLs must be stored using XML, **NNNN;**, encoding. Unicode characters in messages stored in org/Pentaho/ui/messages.properties that are used in XSLs should be accessed using msg:getXslString and output escaping should be disabled.

```
<xsl:value-of
    select="msg:getXslString('UI.USER_FILTER_PANEL_HINT')"
    disable-output-escaping="yes"
/>
```

JFreeReport Definitions

Unicode characters in report definitions for JFreeReport should be stored using XML, **NNNN;**, encoding. In addition the report configuration section should specify that Identity-H encoding and embedded fonts should be used for PDF output.

```
<configuration>
    <property
name="org.jfree.report.modules.output.pageable.pdf.Encoding">Identity-H</property>
    <property name="org.jfree.report.modules.output.pageable.pdf.EmbedFonts">true</property>
</configuration>
```

You need to use fonts that support Unicode characters such as 'Ariel Unicode MS'

08. Submitting Your Translation

This page last changed on Mar 13, 2007 by phart.

07. Unicode Support Internationalization Guide

Email communityconnection@pentaho.org when you are ready to submit any files. You will need to sign a Contributor's Agreement (http://community.pentaho.org/contributor_agreement) and we will send you instructions for how to submit.