Zaval JRC Editor
Version 1.2

User's Guide

Zaval Creative Engineering Group

http://www.zaval.org

Contents

Introduction to the Zaval JRC Editor

Since JDK 1.1.x. Sun introduced Internationalization concept for Java.

> *Internationalization* is the process of designing an application so that it can be adapted to various languages and regions without engineering changes.

> *Localization* is the process of adapting software for a specific region or language by adding locale-specific components and translating text.

> *The Java™ Tutorial*

The text used in programs is locale dependent, that's why we store all text strings in the separate files as a key-value pairs. When we need to localize a program for yet another language support we have to translate values in these files to the desired language. And that's all! These files are called *resources*. We provide GUI tool to manipulate these resources. The files being produced by this tool is fully compatible with ResourceBundle expectations.

So, the Zaval JRC Editor (also known as Zaval Java Resource Editor) solution is small and easy-to-use visual editor for resource files.

What Can You Do with the Zaval Java Resource Editor

The Zaval Java Resource Editor can be used for new and existing software localization, resources synchronization and any other resources manipulations.

It provides full support for any language specific resources (it depends on the fonts and font metrics settings of the host OS at your computer). The target of this tool is localization strings manipulation for all Java-based software that has appropriate support embedded.

When To Use the Zaval Java Resource Editor

The Zaval JRC Editor is best used for regular access to various resource files. You can add your own language support to the existing software if strings are not hard-coded to the software. One of the greatest things in the internationalization is that you don't need to make code changes.

Another great area in this tool usage is resource bundle synchronization. Our tool can handle this task easily – it compares the files set and highlights all differences. It allows separating development process and resource management process. That's why this tool can be used as part of your software pack to provide 3rd party localization.

Internationalization concept overview

Internationalization means that developers can customize their product for different languages and locales. In Java language terms this task handles by ResourceBundle class (see The Java™ Tutorial). All resource files' names have two parts: resource name and localization suffix. When you want your program to 'speak' French, for example, the only thing you need to do is adding corresponding properties file. This procedure does not require any interaction with the software creator.

For more info on the localization see corresponding tutorial at http://java.sun.com.

Installing the Zaval Java Resource Editor

We did our best to make the installation procedure of the product easy, so just follow the instructions.

Requirements

To use the Zaval JRC Editor you need a JDK/JRE installed on your computer. All Sun's JDK (versions since 1.1.7) can be used without any problems. We suggest you to use Sun's JDK 1.2.2 or later as most stable (can be found at http://java.sun.com). Alternatively you can use any other vendors' JDK (IBM's were ok for versions 1.2.x, 1.3.x and 1.4).

Installation procedure

1. Unpack zip/tarball with Zaval JRC Editor to an empty directory.

2. If you are using JDK 1.2 or above start the batch file (editor.bat) to run Zaval Java Resource Editor, otherwise go to step 3.

3. For any other non-standard or old versions of JDK/JRE you need to make several changes to the batch file (or you need to setup environment variables):

   ```
   set JAVA_HOME=<location of JDK>
   set PATH=%JAVA_HOME\jre\bin;%PATH%
   set CLASSPATH=.;%JAVA_HOME%\jre\lib\rt.jar;%CLASSPATH%
   ```

   The last line can have variation for old versions of JDK, such as 1.0.2 or 1.1.x. In this case you need specify the following line instead:

   ```
   set CLASSPATH=.;%JAVA_HOME%\jlib\classes.zip;%CLASSPATH%
   ```

   When you complete the steps above the Zaval JRC Editor is ready to use.

Using Zaval JRC Editor

Application start

To start this application you need simply start corresponding batch file. When the application is started you are able to manipulate resource files.
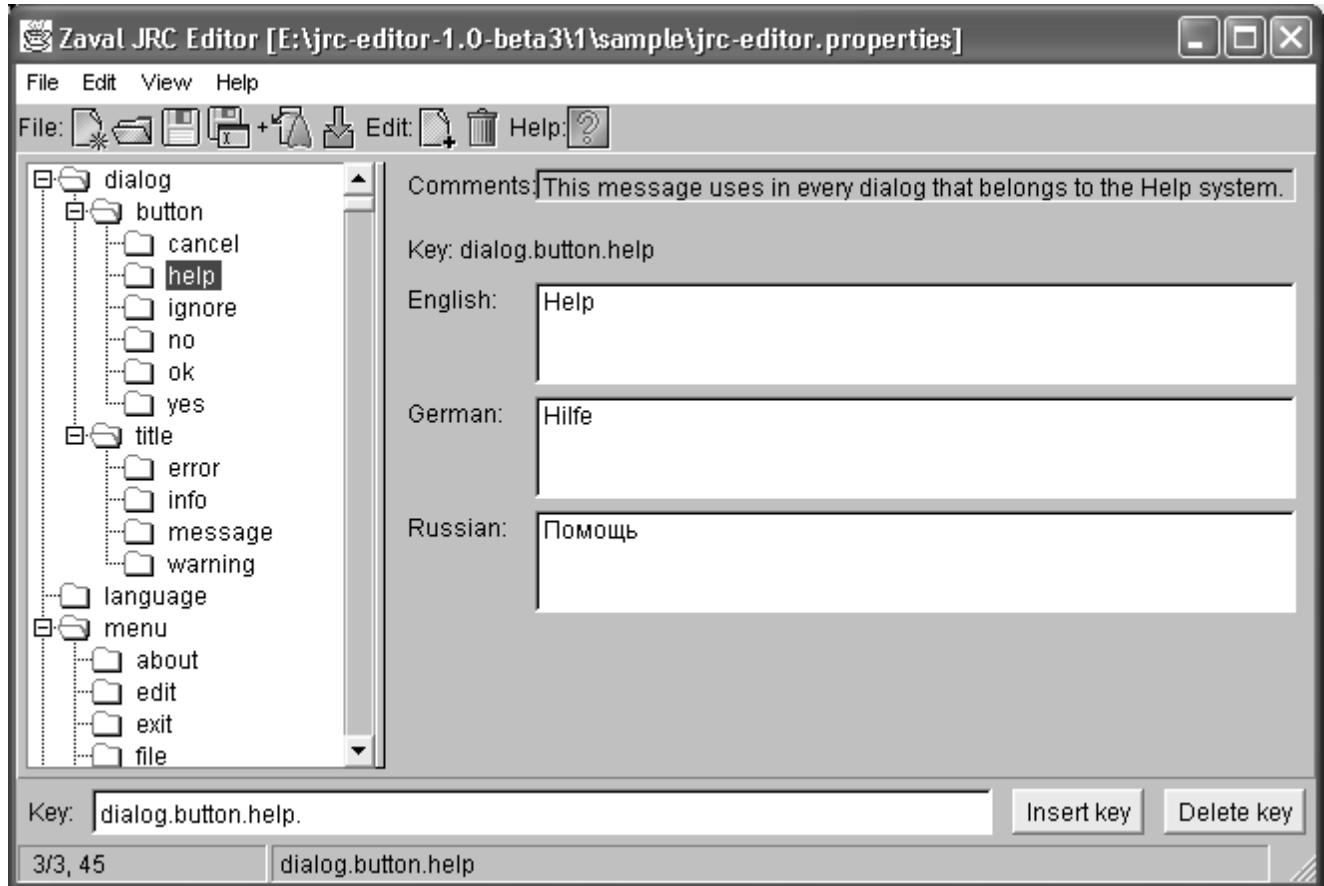
Editing existing resources

Zaval Java Resource Editor consists of two panels displayed on Screenshot 1. To start working with your resource set you should make several steps:

- From the **File** menu, choose **Open** or **New resource**.

- In the opened dialog select resource file that meets mentioned requirements. **Note:** the files previously opened will be closed automatically.

Left panel is a key hierarchy that was stripped out from the appropriate resource files set. The key hierarchy is created automatically using "." or "_" as a separator. So, adding "." in key name automatically creates new hierarchy level.

Right panel displays a list of the languages defined by the resource files set.

Screenshot 1. Working area of the Zaval Java Resource Editor

To hide a language from the right panel go through the following steps:

- From the **View** menu, select **Show resource**.

- In the **Show resource** list, clear a checkbox next to the language.

To edit key property for any language available do the following:

- In the left panel select item you want to edit - it will appear in the **Key** field at the bottom of the panel (in our example **dialog.button.help**).

- In the right panel make necessary changes to the corresponding field (see Screenshot 1).

To apply changes use **File -> Save** from the top menu. Resource file's name consists of base name (e.g. **'editor'**), language suffix (e.g. **'de'**), and **.properties** extension, for example, **editor_de.properties** (German), **editor_fr.properties** (French). In our

example **jrc-editor.properties** file contains default English messages and is displayed in the title of the Zaval Java Resource Editor's panel.

To save all files with a different base name and/or write them to a different directory, use **File -> Save as** instead.

Creating new resources

There are several ways to create new resources: either enter them by hands, or importing them from *.java files. For more info see **Importing resources from *.java files** section.

Inserting new items

To add your own items to the tree proceed through the following steps:

- In the **Key** field, type tree key, for example **dialog.button.new**.

- Press **Insert** button.

- Fill in all Key's values.

- Repeat these steps for all items you want to add.

Deleting items
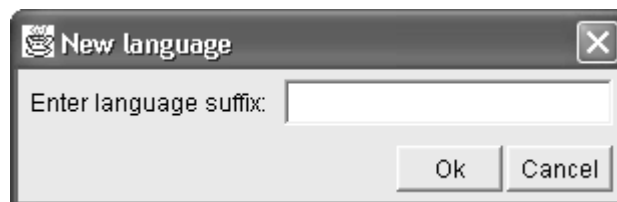
The remove a key-value pair do the following:

- Select the appropriate key at the left panel.
- Press **Delete** button.
- You will be asked either you want to remove this key with all sub-keys or this key only.

Note: removing items from the .properties file may cause application to fail. Think over before you proceed. Unused items will do no harm to the application stability.
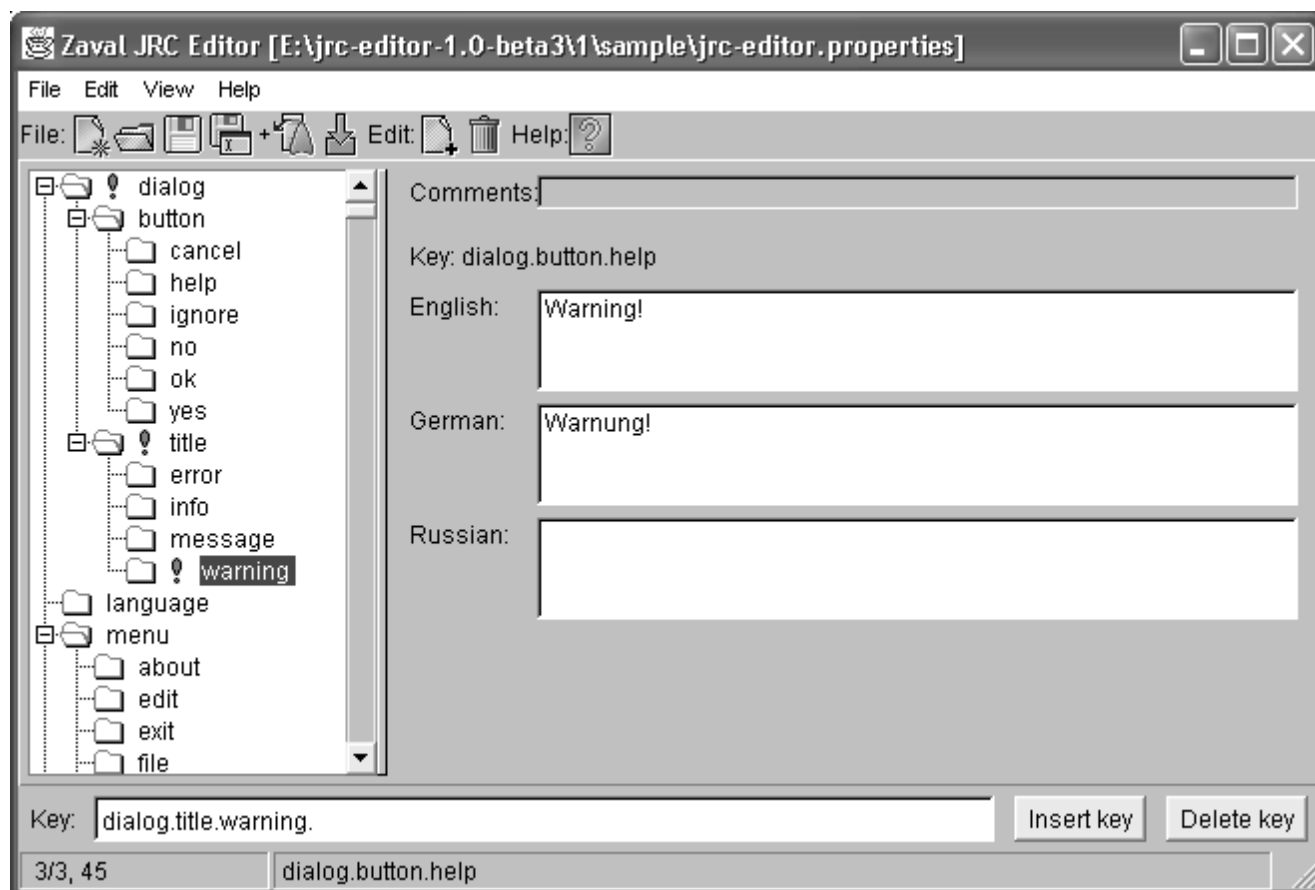
Adding new language

Adding new language is simple:

- Choose **Edit -> New language**.

- In the **New resource** dialog (see Screenshot 2), enter resource file suffix, for example, **fr**. When you click OK corresponding .properties file will be created (in our example **editor_fr.properties**) and edit fields for French language will appear in the right panel.

Screenshot 2. Adding new language

An exclamation mark next to tree node notifies you that there is at list one item in this node that is not translated into all languages displayed in the right panel (see Screenshot 3).

Screenshot 3. Resources synchronization

When edit field next to a language is left empty, then it defaults to English language.

Tree manipulations

To make your life easier when working with really big files we added expand/collapse functionality to whole tree and any of its nodes – see **Tree** menu.

Import/Export

Starting from version 1.2 JRC Editor supports data import/export from single XML file or single plaintext file. Format is pretty simple, so there shouldn't be any problems.

Fast navigation

To make your life easier we have added shortcuts support for all important actions. For example: Save – Ctrl+S, Open – Ctrl+O, etc.
Starting from version 1.2 you can access all JRC Editor functions without a mouse support – you can navigate through all JRC Editor controls with Tab key, access upper menu functions using shortcuts and more.

Importing resources from *.java files

To simplify existing applications transfer to the resources usage you can use **Parse source...** feature. It goes through java sources, extracts text strings and builds tree-like structure based on class name/package name. Now you can import resources only from one file at a time and can't do iterations.

Importing process doesn't get all strings but only those that look right: for example, it wouldn't import any single character strings.

Resource stencil source code generation

Stencil is a java source that contains all resource strings with their initialization and corresponding getters/setters. Initialization requires appropriate resource bundle.
To generate use **File -> Generate source code...**.

Null keys

Null keys or blank keys are keys that have no value for all languages. Often it can be useful to see all of them (see **View -> Indicate blank resources**).

Statistics

Sometimes it is useful to see statistics over files you are working with, so we added this functionality (see **View -> Statistics**).

Product limitations

There are several product limitations that can be fixed in future versions:

- You can't operate with multiple resource sets – only one resource set available at a time.

- You cannot operate with resources in JAR files.

- You cannot analyze a lot of Java sources at a time.

- You cannot split all lines being generated to multiple targets (Java files).

Further product plans

Current tool implementation follows the minimalist computing concept. In near future the following features will be added:

- Ability to merge different resource sets;

- Ability to split one existing resource set to multiple ones;

- Code generation and analysis improvements and multiple operations support;

- Compressed resource set support (*.jar);

Support available

All support for software installation and problems should be sent directly to support@zaval.org with "Re: Zaval JRC Editor Support" in subject line and plain text in the message body, describing your request and/or your problem. Since this software is distributed under the General Public License and is maintained by its authors on non-

commercial basis, your request will be answered as soon as possible, but no later than 5 business days.

The Zaval Creative Engineering Group carries out its software customization/new software development on the regular basis. For more info contact us at info@zaval.org.

Stay informed!

Now you can receive information on latest products' updates and hotfixes via email. This is a low-traffic list (1-2 messages per month). To subscribe, send blank mail to news-subscribe@zaval.org.