Sun Directory Server Enterprise Edition 7.0 Localization Guide

December 2009

This document explains how to organize the Sun™ Directory Server Enterprise Edition software resource files, add support for new languages, and customize the existing software resources.

Sun Directory Server Enterprise Edition is currently localized into five languages: French, German, Japanese, Spanish, and Simplified Chinese. Localizing the administration browser interface (DSCC) and the command-line interface are the main requirements for supporting a new language that is not part of the Sun Directory Server Enterprise Edition Product.

This technical note describes the entire localization process, including which files store localized messages, where these files are located, and how to add a new language to Sun Directory Server Enterprise Edition.

This document is applicable only to Sun Directory Server Enterprise Edition version 7.0.

This document contains the following sections:

- "Organization of the Resource Files" on page 2
- "Adding New Language Support to the Directory Service Control Center" on page 4
- "Adding New Language Support to the Command Line Interface" on page 5

Organization of the Resource Files

This section describes the organization of the files that contain Sun Directory Server Enterprise Edition messages.

The application messages are stored in the following file formats:

- `.properties` files (Java properties) as jar files
- `.res` files (ICU resource text files)

Messages that are displayed in the command-line interface are stored in both of these file formats. Messages that are displayed in the Directory Service Control Center (web interface) are stored only in Java properties file format (`.jar` files).

All of these files are compiled and delivered as native packages (for Solaris) and as zip files (for all platforms). The resource files can be easily modified to add new locales or according to the application requirements. The following sections describe the organization of the software resource files.

Note – Information about file organization is accurate up to and including Sun Directory Server Enterprise Edition 7.0.

Java Properties Files

The Java properties files contain most of the software messages that are displayed in the Directory Service Control Center (DSCC) Administration Console. Java properties files should be saved in escaped Unicode format, otherwise, multi-byte characters are displayed incorrectly.

To ensure the correct format of the Java properties files, type the following command after each change:

`# native2ascii` *source-filename destination-filename*

For more information about the `native2ascii` command, see `http://java.sun.com/j2se/1.5.0/docs/tooldocs/#intl`.

Handling Special Characters

In the Java properties files, apostrophes (') (ASCII 39) in the translated messages must be doubled. For example, the French translation of the following string:

`"Error while setting access rights on NSS database"`

would be as follows:

```
"Erreur lors de la définition des droits d''accès sur la base de donnies NSS"
```

The `Resources.properties` file is an exception to this rule. Apostrophes in this file must only be doubled when the message contains a parameter in the form "{n}", where n is a number from 0 to 9. In this case the translation of the following string:

```
"Error reading installation configuration"
```

would be as follows:

```
"Erreur lors de la lecture de la configuration de l'installation"
```

However, the translation of the following string:

```
"Install error Code: {0}<br /><br />Stack:<br />{1}"
```

would be as follows:

```
"Code d''erreur d''installation : {0}<br /><br />Pile:<br />{1}"
```

To avoid problems with apostrophes, you can use the right single quotation mark unicode character (') (U+2019) instead. Note, however, that this character does not display correctly in the Microsoft Windows command-line interface.

ICU Resource Files

ICU Resource Bundles are the resource files that contain most of the software messages that are displayed in Directory Server and Directory Proxy Server, and common messages in the native code. As is the case for the Java properties files, you should convert the ICU files into an escaped Unicode format to prevent multi-byte characters from being displayed incorrectly.

To ensure the correct format of ICU text files, type the following command after each change:

```
# native2ascii source-filename destination-filename
```

For more information about the `native2ascii` command, see http://java.sun.com/j2se/1.5.0/docs/tooldocs/#intl.

ICU Resource Bundle text files must be compiled into `.res` files by using the `genrb` command. The `genrb` command is platform-dependent, that is, the resulting `.res` files are only usable on the platform on which they were created.

The `genrb` command is used as follows:

```
# genrb [--sourcedir ${sourcedir}] [--destdir ${tempdir}] SOURCE_FILE
```

For information about the `genrb` command, and to download the appropriate version of the command, see http://www.icu-project.org/userguide/localizing.html and http://www.icu-project.org/userguide/ResourceManagement.html.

Handling Special Characters

In the ICU resource files, all apostrophes (') (ASCII 39) in the translated messages must be doubled. See the previous section for examples.

In the ICU resource files, the double quote character (") (ASCII 34) must be escaped by a backslash. For example, the following string:

```
"Using \"{0}\" as default value for {1} option"
```

would be translated into French as follows:

```
"Utilisation de \"{0}\" comme valeur par défaut pour l''option {1}"
```

Adding New Language Support to the Directory Service Control Center

The Directory Service Control Center (DSCC) is the web-based administration console for Sun Directory Server Enterprise Edition. This chapter shows the basic steps involved in adding new language support to the DSCC and customizing the Sun Directory Server Enterprise Edition localization resource files.

File Structure and Naming Conventions

The files that contain messages are located in the following directory:

directory-installation-path/resources/dcc7app/WEB-INF/lib/dccapp.jar

The common UI resource catalog is located in websdk/lib/cc_*locale*.jar (from the file *directory-installation-path*/lib/web/websdk.zip).

The Java properties file naming conventions are as follows:

com/sun/web/admin/directory/dcc/resources/Resources_*locale*.properties

and

com/sun/web/ui/resources/Resources_*locale*.properties

Adding New Language Support to the DSCC User Interface

The following procedure describes how to add a new language to the DSCC user interface.

▼ To Add a New Language to the DSCC User Interface

1 Unregister Directory Service Control Center as follows:
directory-installation-path/bin/dsccsetup console-unreg

2 Stop Application Server as follows:
application-server-installation-path/bin/asadmin stop-domain -user admin domain1

3 Copy the dccapp.jar **and** cc.jar **files to a temporary directory.**

   a. Locate dccapp.jar **in the following directory:**
   directory-installation-path/resources/dcc7app/WEB-INF/lib/

   b. Unzip the file *directory-installation-path*/lib/web/websdk.zip**.**

   c. Locate cc.jar **in the following directory:**
   directory-installation-path/lib/web/websdk/lib/

4 Unjar the dccapp.jar **and** cc.jar **files.**
The two source files to be translated are as follows:
com/sun/web/admin/directory/dcc/resources/Resources.properties
com/sun/web/ui/resources/Resources.properties

5   Create the corresponding translated Java properties files named (`Resources_locale.properties`) **in the appropriate directories.**

For example, create Dutch language resource files as follows:

```
com/sun/web/admin/directory/dcc/resources/Resources_nl.properties
com/sun/web/ui/resources/Resources_nl.properties
```

6   Convert the files into escaped Unicode by typing the following command:

```
# native2ascii source-filename destination-filename
```

7   Add the resource catalogs to the .jar file.

For example, to add the `Resources_nl.properties` file under `com/sun/web/admin/directory/dcc/resources/Resources_nl.properties`, type the following command:

```
# jar -uvf directory-installation-path/resources/dcc7app/WEB-INF/lib/dccapp.jar \
temporary-directory
```

Create a localized `cc.jar` file as follows:

```
# jar -cvf cc_nl.jar temporary_directory
```

The file should have the below structure:

```
temporary_directory/META-INF/
temporary_directory/META-INF/MANIFEST.MF
temporary_directory/com/sun/web/ui/resources/Resources_nl.properties
```

8   Create a .war file with all of the contents and save it in the following location:

```
directory-installation-path/var/dscc7.war
```

```
# jar -cvf dscc7.war temporary_directory
```

9   Deploy the web application in the Application Server by using the following command:

```
application-server-installation-path/bin/asadmin start-domain -user admin domain1
```

10   Set your browser preferred language and log in to DSCC.

Adding New Language Support to the Command Line Interface

This chapter shows the basic steps involved in adding new language support to the command line interface and customizing the Sun Directory Server Enterprise Edition localization resource files.

Structure and Naming Conventions of Java Properties Files

The CLI interface messages are located in the following directories:

```
directory-installation-path/lib/jar/slapwcli_l10n.jar
directory-installation-path/lib/jar/slapycli_l10n.jar
directory-installation-path/lib/jar/slaptcli_l10n.jar
directory-installation-path/lib/jar/dpcfgcli_l10n.jar
directory-installation-path/lib/jar/dpcfg_l10n.jar
directory-installation-path/lib/jar/common_cfg_l10n.jar
directory-installation-path/lib/jar/clip_l10n.jar
```

The Java properties files are named *filename_locale*.`properties` and include the following files:

```
directory-installation-path/lib/jar/messages/slapw_locale.properties
directory-installation-path/lib/jar/messages/slapy_locale.properties
```

directory-installation-path/lib/jar/messages/slapt_*locale*.properties
directory-installation-path/lib/jar/messages/dpcfg_cli_*locale*.properties
directory-installation-path/lib/jar/messages/dpcfg_context_*locale*.properties
directory-installation-path/lib/jar/messages/dpcfg_core-context_*locale*.properties
directory-installation-path/lib/jar/messages/dpcfg_generic_*locale*.properties
directory-installation-path/lib/jar/messages/common_cfg_*locale*.properties
directory-installation-path/lib/jar/messages/clip_*locale*.properties

Adding New Language Support to the CLI for Java Properties

The following procedure describes how to add a new language to the command line interface.

▼ To Add a New Language to the Command Line Interface for Java Properties

1   Locate the .jar **files listed in the previous section and copy them to a temporary directory.**

2   Stop all Directory Server and Directory Proxy Server instances that are running as follows:
directory-installation-path/bin/dsadm stop-running-instances
directory-installation-path/bin/dpadm stop-running-instances

3   To add a new language to the slapwcli_l10n.jar **resources catalog file, do the following:**

    a. Unjar the *directory-installation-path*/lib/jar/slapwcli.jar **file to a temporary directory.**

    b. Translate the English language file, messages/slapw.properties**, into the required language.**

    c. Move the translated file to slapw_*locale*.properties **in the messages directory.**
      For example, if you are creating a Dutch language resource file, move it to
      messages/slapw_nl.properties.

    d. Convert the file into escaped Unicode by using the following command:
      # native2ascii *source-filename destination-filename*

    e. Add the resource file to slapwcli_l10n.jar **by typing the following command:**
      # jar -uvf slapwcli_l10n.jar *temporary_directory*

    f. Replace the file *directory-installation-path*/lib/jar/slapwcli_l10n.jar **with your new .jar file.**

4   To add a new language to the slapycli_l10n.jar **resources catalog file, do the following:**

    a. Unjar the *directory-installation-path*/lib/jar/slapycli.jar **file to a temporary directory.**

    b. Translate the English language file, messages/slapy.properties**, into the required language.**

    c. Move the translated file to slapy_*locale*.properties **in the messages directory.**
      For example, if you are creating a Dutch language resource file, move it to
      messages/slapy_nl.properties.

    d. Convert the file into escaped Unicode by using the following command:
      # native2ascii *source-filename destination-filename*

    e. Add the resource file to slapycli_l10n.jar **by typing the following command:**
      # jar -uvf slapycli_l10n.jar *temporary_directory*

    f. Replace the file *directory-installation-path*/lib/jar/slapycli_l10n.jar **with your new .jar file.**

5   To add a new language to the `slaptcli_l10n.jar` **resources catalog file, do the following:**

    a. **Unjar the** *directory-installation-path*/`lib/jar/slaptcli.jar` **file to a temporary directory.**

    b. **Translate the English language file,** `messages/slapt.properties`, **into the required language.**

    c. **Move the translated file to** `slapt_`*locale*`.properties` **in the messages directory.**

        For example, if you are creating a Dutch language resource file, move it to `messages/slapt_nl.properties`.

    d. **Convert the file into escaped Unicode by using the following command:**

        `# native2ascii `*source-filename* *destination-filename*

    e. **Add the resource file to** `slaptcli_l10n.jar` **by typing the following command:**

        `# jar -uvf slaptcli_l10n.jar `*temporary_directory*

    f. **Replace the file** *directory-installation-path*/`lib/jar/slaptcli_l10n.jar` **with your new .jar file.**

6   To add a new language to the `dpcfgcli_l10n.jar` **resources catalog file, do the following:**

    a. **Unjar the** *directory-installation-path*/`lib/jar/dpcfgcli.jar` **file to a temporary directory.**

    b. **Translate the English language file,** `messages/cli.properties`, **into the required language.**

    c. **Move the translated file to** `cli_`*locale*`.properties` **in the messages directory.**

        For example, if you are creating a Dutch language resource file, move it to `messages/dpcfg/cli_nl.properties`.

    d. **Convert the file into escaped Unicode by using the following command:**

        `# native2ascii `*source-filename* *destination-filename*

    e. **Add the resource file to** `dpcfgcli_l10n.jar` **by typing the following command:**

        `# jar -uvf dpcfgcli_l10n.jar `*temporary_directory*

    f. **Replace the file** *directory-installation-path*/`lib/jar/dpcfgcli_l10n.jar` **with your new .jar file.**

7   To add a new language to the `dpcfg_l10n.jar` **resources catalog file, do the following:**

    a. **Unjar the** *directory-installation-path*/`lib/jar/dpcfg.jar` **file to a temporary directory.**

    b. **Obtain the three English language files (**`context.properties`, `core-context.properties`, **and** `generic.properties`**) in the directory** *directory-installation-path*/`lib/jar/dpcfg.jar` **and translate them into the required language.**

    c. **Move the translated files to** `context_`*locale*`.properties`, `core-context_`*locale*`.properties`, **and** `generic_`*locale*`.properties` **in the** `messages` **directory.**

        For example, if you are creating a Dutch language resource file, move them to the following files:

        `messages/dpcfg/context_nl.properties`
        `messages/dpcfg/core-context_nl.properties`
        `messages/dpcfg/generic_nl.properties`

    d. **Convert the files into escaped Unicode by using the following command:**

        `# native2ascii `*source-filename* *destination-filename*

    e. **Add the resource file to** `dpcfg_l10n.jar` **by typing the following command:**

        `# jar -uvf dpcfg_l10n.jar `*temporary_directory*

f.   **Replace the file** *directory-installation-path*/`lib/jar/dpcfg_l10n.jar` **with your new .jar file.**

8   **To add a new language to the** `common_cfg_l10n.jar` **resources catalog file, do the following:**

a.   **Unjar the** *directory-installation-path*/`lib/jar/common_cfg.jar` **file to a temporary directory.**

b.   **Translate the English language file,** `common_cfg.properties`, **into the required language.**

c.   **Move the translated file to** `common_cfg_`*locale*`.properties` **in the messages directory.**
     For example, if you are creating a Dutch language resource file, move it to
     `messages/common_cfg_nl.properties`.

d.   **Convert the files into escaped Unicode by using the following command:**
     `# native2ascii` *source-filename destination-filename*

e.   **Add the resource file to** `common_cfg_l10n.jar` **by typing the following command:**
     `# jar -uvf common_cfg_l10n.jar` *temporary_directory*

f.   **Replace the file** *directory-installation-path*/`lib/jar/common_cfg_l10n.jar` **with your new .jar file.**

9   **To add a new language to the** `clip_l10n.jar` **resources catalog file, do the following:**

a.   **Unjar the** *directory-installation-path*/`lib/jar/clip.jar` **file to a temporary directory.**

b.   **Translate the English language file,** `clip.properties`, **into the required language.**

c.   **Move the translated file to** `clip_`*locale*`.properties` **in the messages directory.**
     For example, if you are creating a Dutch language resource file, move it to
     `messages/clip_nl.properties`.

d.   **Convert the file into escaped Unicode by using the following command:**
     `# native2ascii` *source-filename destination-filename*

e.   **Add the resource file to** `clip_l10n.jar` **by typing the following command:**
     `# jar -uvf clip_l10n.jar` *temporary_directory*

f.   **Replace the file** *directory-installation-path*/`lib/jar/clip_l10n.jar` **with your new .jar file.**

ICU Resource Files

The command line interface messages are located in the following directories:

- *directory-installation-path*/`resources/dsadmin/dsadmin_`*locale*`.res`
- *directory-installation-path*/`resources/dpadm/dpadm_`*locale*`.res`

The ICU resource files have the following naming convention: `filename_`*locale*`.res`.

Adding New Language Support to the CLI for ICU Resource Files

The following procedure describes how to add a new language to the command line interface for ICU
resource files.

▼ To Add a New Language to the Command Line Interface for ICU Resource Files

1 Request the ICU files for translation from the global alias directory-interest@sun.com.

2 Stop all Directory Server and Directory Proxy Server instances that are running as follows:

directory-installation-path/bin/dsadm stop-running-instances
directory-installation-path/bin/dpadm stop-running-instances

3 To add a new language to the ICU resource catalog files, do the following:

a. Obtain and translate the following files:

```
tmpl_dsadmin_en.txt
tmpl_dpadm_en.txt
common_adm_en.txt
```

b. Rename the translated files with the suffix that corresponds to your locale.

For example, to create Dutch language resource files, rename the files as follows:

```
tmpl_dsadmin_n1.txt
tmpl_dpadm_n1.txt
common_adm_n1.txt
```

c. Convert the files into escaped Unicode by using the following command:

native2ascii *source-filename destination-filename*

d. Combine the files into the correct ICU format as follows:

echo "*locale* {" > dsadmin_*locale*.txt
cat tmpl_dsadmin_*locale*.txt >> dsadmin_*locale*.txt

cat common_adm_*locale*.txt >> dsadmin_*locale*.txt
echo "}" >> dsadmin_*locale*.txt

echo "*locale* {" > dpadm_*locale*.txt
cat tmpl_dpadm_*locale*.txt >> dpadm_*locale*.txt
cat common_adm_*locale*.txt >> dpadm_*locale*.txt
echo "}" >> dpadm_*locale*.txt

e. Compile the files by using the genrb **command.**

genrb dsadmin_*locale*.txt
mv *locale*.res *directory-installation-path*/resources/dsadmin/dsadmin_*locale*.res
genrb dpadm_*locale*.txt
mv locale.res *directory-installation-path*/resources/dpadm/dpadm_*locale*.res

Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A.