

MorphoAccess® SIGMA Family

Application note - Message Translation



COPYRIGHT© 2017 IDEMIA

Osny, France

Warning

COPYRIGHT© 2017 IDEMIA. All rights reserved.

Information in this document is subject to change without notice and do not represent a commitment on the part of IDEMIA. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying or recording, for any purpose without the express written permission of IDEMIA.

This legend is applicable to all pages of this document.

This manual makes reference to names and products that are trademarks of their respective owners.

Revision History

The table below contains the history of changes made to the present document.

Version	Date	Description
01	November 2013	First version
02	April 2018	Update company name (IDEMIA) Add custom font support
03	May 2019	Add CUSTO JSON file support for webserver language

Application Note

MorphoAccess® SIGMA Series: Messages Translation

Introduction

This note applies to MorphoAccess® SIGMA Series terminals. These terminals are dedicated to physical access control using biometrics and contactless cards.

This note explains how to create a new set of messages, usually for a language not already provided with the terminal. This note applies only to terminal with a screen to display messages to users and to administrators.

Limits

The current note explains how to use **Qt Linguist** application to translate English messages into any another language, such as Latin languages and Arabic.

This solution has been evaluated with the version 4.6, 4.8 and 5.1.1 of Qt Linguist.

IDEMIA provides support only for the usage of **Qt Linguist** to translate MorphoAccess® SIGMA Series messages. If support is required for another use, please consult Qt user's forum.

References

Where to download a standalone version (**Qt Linguist** application only):

<http://qt-apps.org/content/show.php?content=89360>

Where to download the latest version (within Qt framework):

<http://qt-project.org/downloads>

Help on Qt and Qt Linguist can be found in Qt forum, here:

<http://qt-project.org/forums>

Documentation can be found here:

<http://qt-project.org/doc/>

Description of translation process (step by step)

Introduction

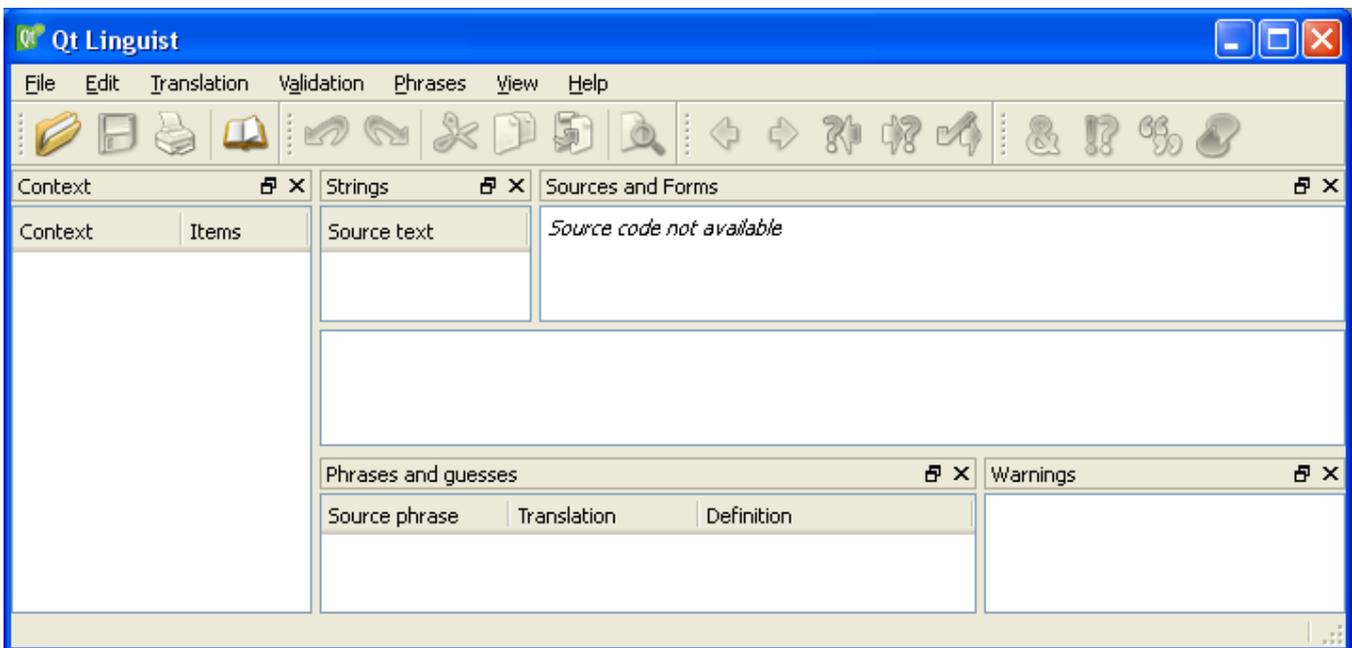
This tutorial describes the translation process, using as sample, the translation of English messages (reference) into Arabic language (target).

Prerequisite

1. **Qt Linguist** application is installed in PC: this application is not provided by IDEMIA, but must be downloaded from Qt developers' web site.
2. English messages file (**English.ts**) must be copied on the PC: this file is provided by IDEMIA on request only.

First Phase: application settings

1. Launch Qt Linguist application. It shows below default screen (Windows sample)



2. Chose "**Open**" in "**File**" menu and select the "**English.ts**" file which contains the English messages (reference messages).
3. At file opening, Qt Linguist application displays a window which enables to select translation options for selection of target country configuration.



4. Don't modify "Source Language" settings
5. Chose "Target language": "Arabic" (sample) and press ok.



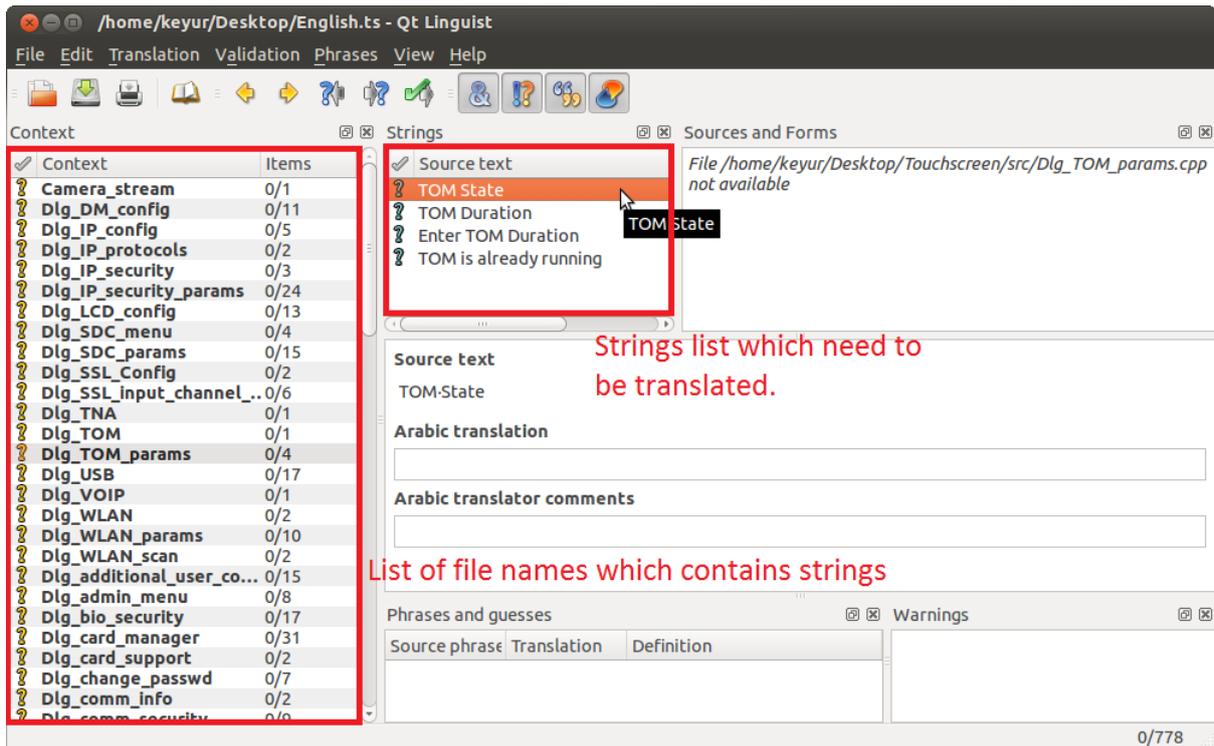
6. Save file as "Arabic.ts" using "Save" command in "File" menu.

Second Phase: Translation of all messages

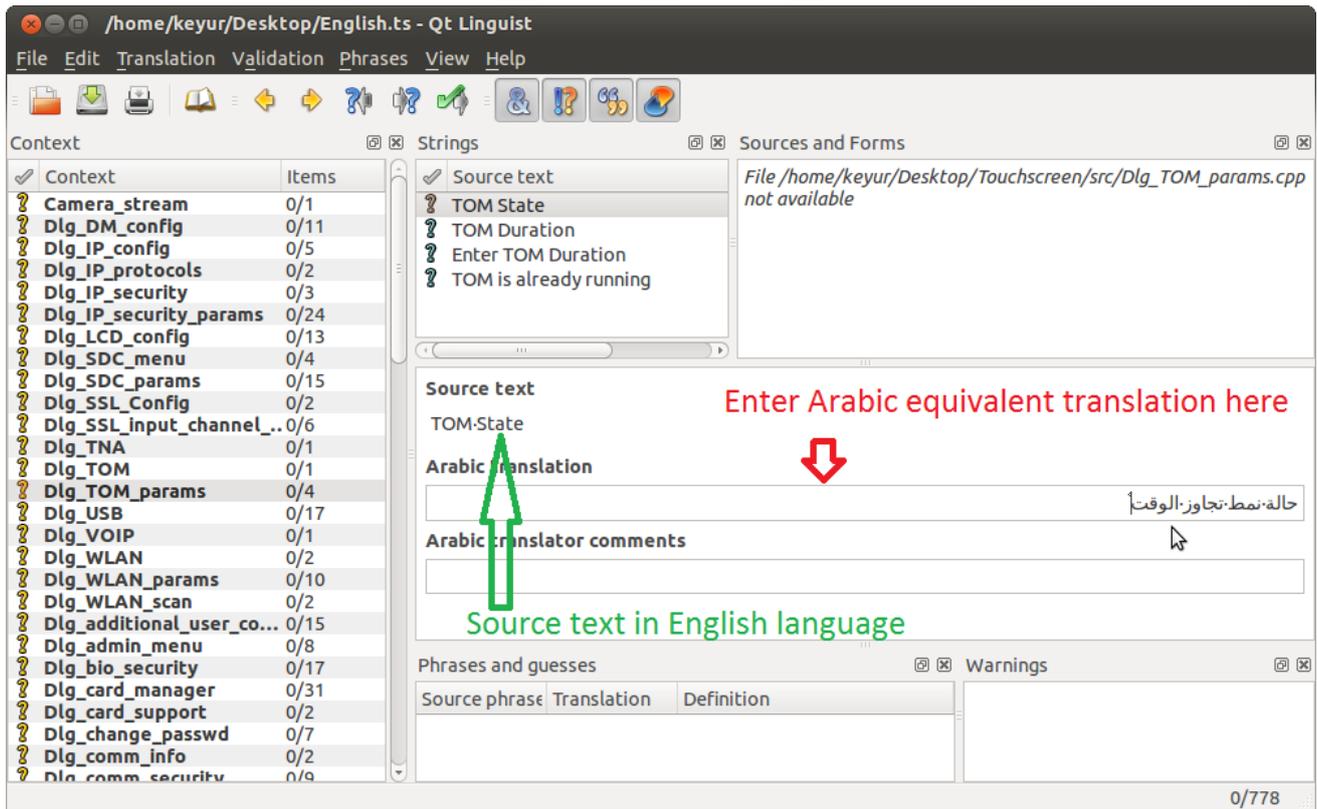
In **context** window all file names (one set of messages per file) are shown with question mark ('?') which is indicating that those string files need to be translated and are not yet translated.

Once those file strings are translated then that symbol will become to green tick mark ('✓').

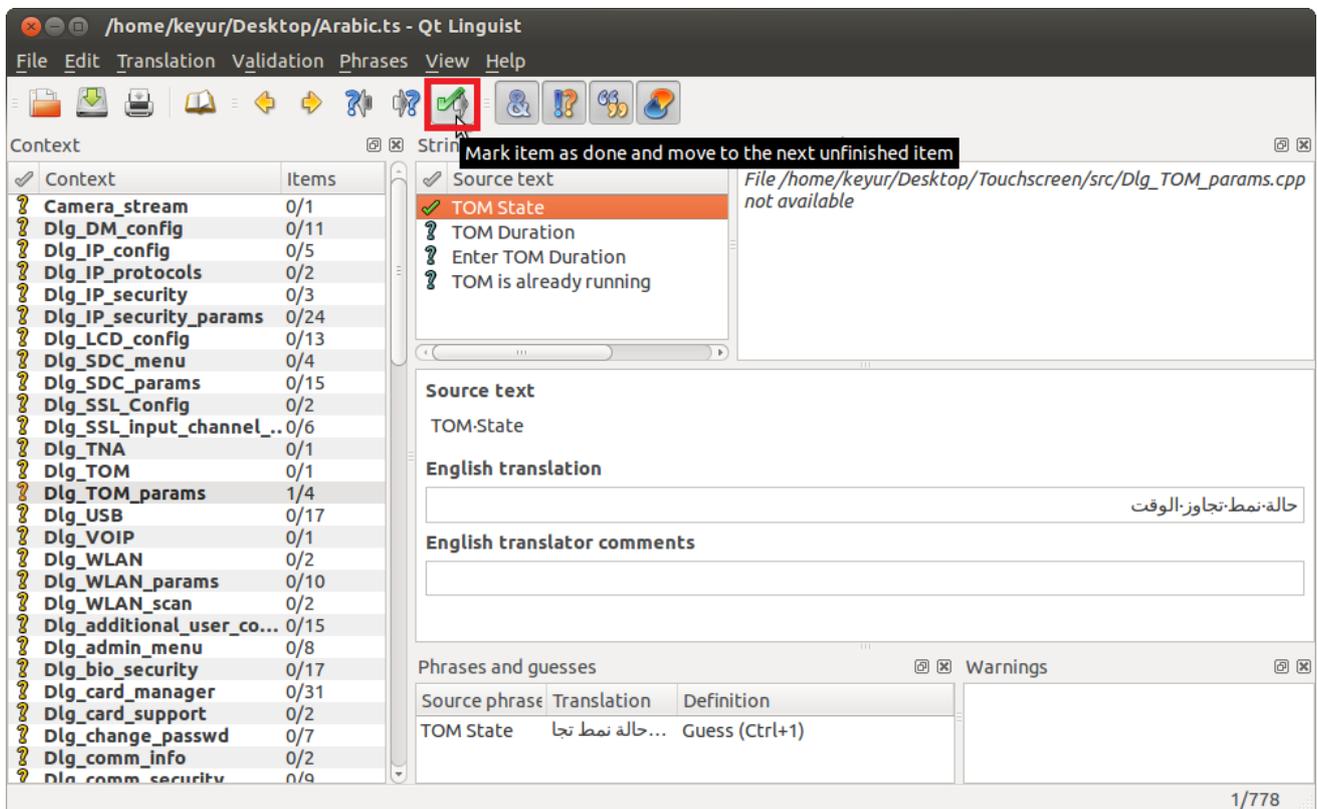
In **string** window strings are displayed for currently selected file from **context** window with same ('?') symbol which needs to be translated.



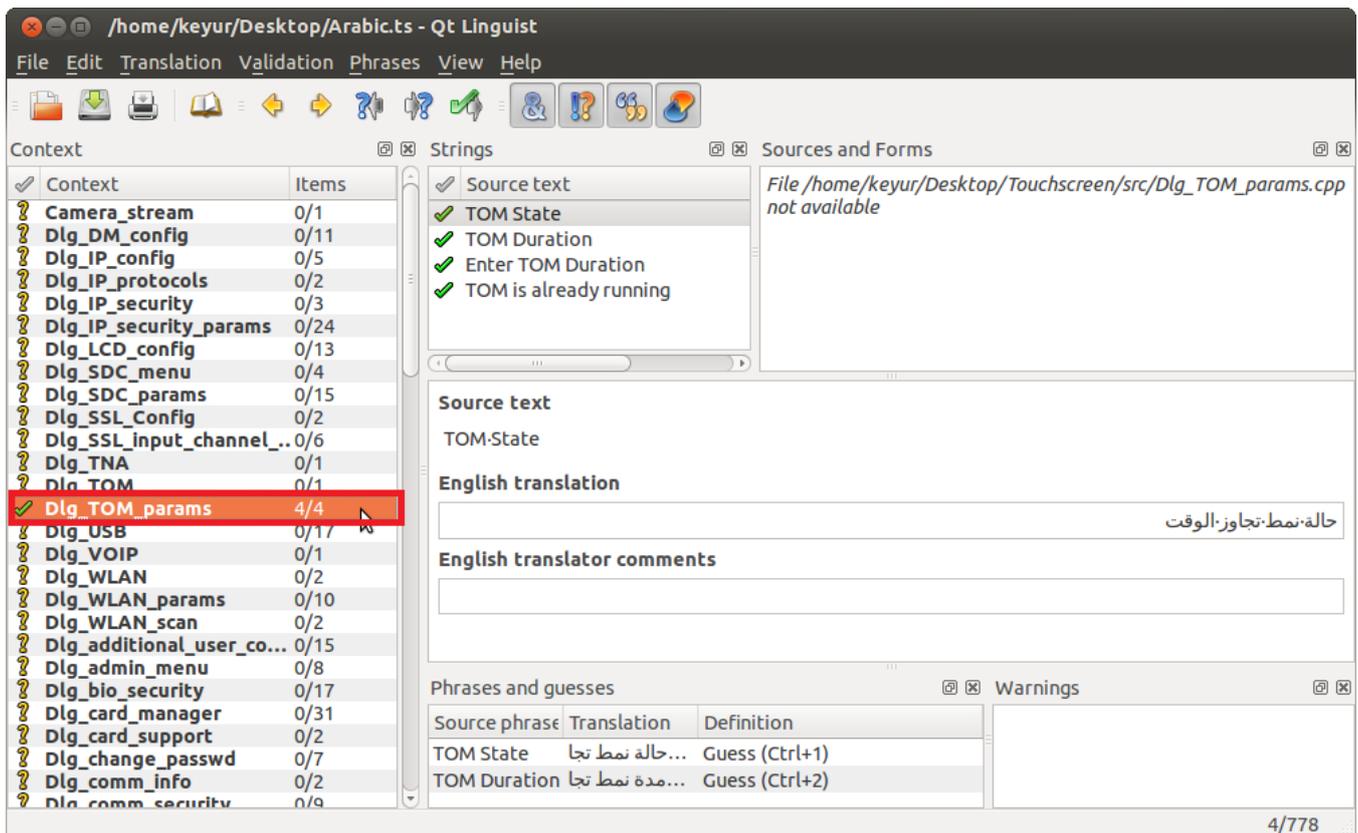
1. First select a **file** in **context** list, and then select a message in **source text** list. In our case we have selected **Dlg_TOM_params** file which contains four strings which we need to translate into Arabic. Now select first **source text** (i.e. "TOM State") and write its equivalent Arabic string in **Arabic translation** box in **Source text** area as mentioned in screen shot below.



- Now click on green tick ('v') button (highlighted in red box on screen shot below) for confirming translation of the string.



- Hence the question mark ('?') icon of selected source string changes to a green tick mark on green tick ('✓'), in **source text** list
- Once all strings of a file are translated then the question mark ('?') icon of the file becomes also a green tick ('✓' as shown in screenshot below (in **Context** list).



- The validation of a translation string doesn't save automatically, the changes must be save manually by a click on the save button (second icon from the left, under "**Edit**" menu). It is recommended to save frequently.

Last phase : generate result file and load it

- Once all strings are translated successfully, it is recommended to create a backup file using "save as" command in "**File**" (select ".ts" format, and add something such as the date to the default name),
- Finally generate a file with the translated messages using "**Release as**" command in **File** menu. Chose a simple name for the .qm file (such as "**Arabic.qm**")
- Upload the .qm file in MorphoAccess® SIGMA terminal. Then the new language is available in language selection menu of the terminal.

Support of custom font

From FW 4.3, the MorphoAccess® SIGMA Series terminals with LCD supports the custom font.

The file format of the custom font is “.ttf”. The upload of the .ttf file is possible via a distant command or via the Morpho Bio Toolbox release 3.4.9.

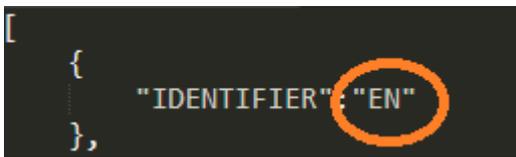
The custom font will be applied only to custom language menus (not to default language) and to user messages (Access granted, Access denied popups)

How to create Custom JSON File

From FW 4.6, the MorphoAccess® SIGMA Series terminals supports the CUSTO JSON file for webserver language. The file format is “.json”. The upload of Custo json file is possible via a distant command or via the Morpho Bio Toolbox.

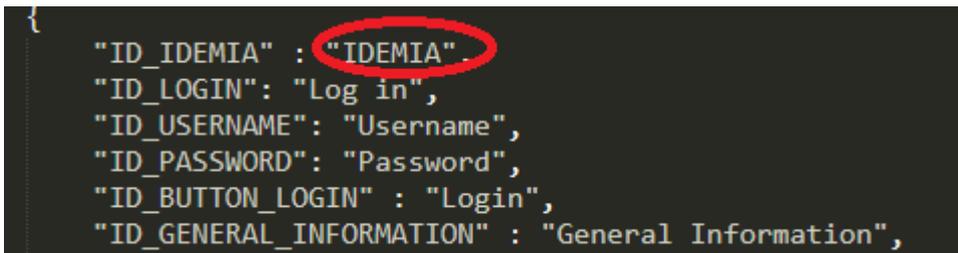
Steps to create CUSTO JSON file:

1. English Custo JSON file (**English.json**) or French Custo JSON file (**French.json**) must be copied on the PC.
2. Open json file in any text editor.
3. Update the “IDENTIFIER” key value in json file by Custo language identifier.



```
[  
  {  
    "IDENTIFIER": "EN"  
  },  
]
```

4. Update the value of each key in json file based on Custom language.



```
{  
  "ID_IDEMIA" : "IDEMIA",  
  "ID_LOGIN" : "Log in",  
  "ID_USERNAME" : "Username",  
  "ID_PASSWORD" : "Password",  
  "ID_BUTTON_LOGIN" : "Login",  
  "ID_GENERAL_INFORMATION" : "General Information",  
}
```

5. Save the file with any name.