[[INTERNE[Only for internal use]]

|  |  | *Screen specification* |
| --- | --- | --- |
|  |  | *i*MX TRANSLATION TOOL DELIVERY CHECK LIST |
| All rights reserved. No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems – without the written permission of the publisher. | | |

|  |  |
| --- | --- |
| File: | imx\_translation\_tool\_delivery\_check\_list\_1.0.4\_EN\_TAG.doc |
| Author: | CODIX |
| Version: | English |

**HISTORY OF VERSIONS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Updated by** | **Modifications** |
| 1.0.4 | 18/09/2018 | P. Gradinarska | P. Gradinarska | Added information about the DCL tables.[[INTERNE[TTS: CUP-23734]] |
| 1.0.3 | 22/06/2016 | P. Gradinarska | P. Shurlieva | Additional information related to the functionalities generating a patch with one or more .res files [[INTERNE[TTS: KBCCFDEV-814 and CODIXDEV-534]] |
| 1.0.2 | 21/10/2015 | P. Gradinarska | P. Gradinarska | Additional note related to the installation of v\_menu, t\_ssmenu and v\_translation tables  [[INTERNE[TTS task: [CSOBDEV-977](https://tts.codix.eu/jira/browse/CSOBDEV-977)]] |
| 1.0.1 | 30/09/2015 | D. Voykova | P. Gradinarska | Additional explanatory info to ‘Prepare for patch’ button  [[INTERNE[TTS task: [SUDFFUP-408](https://tts.codix.eu/jira/browse/SUDFFUP-408) ]] |
| 1.0.0 | 23/09/2015 | P. Gradinarska | P. Gradinarska | Creation [[INTERNE[TTS : [CUP-15738](https://sauron.imxbg/jira/browse/CUP-15738) ]] |

List of contents

[**HISTORY OF VERSIONS** 2](#_Toc454362091)

[List of contents 3](#_Toc454362092)

[1 DOCUMENT DESCRIPTION 4](#_Toc454362093)

[1.1 Scope 4](#_Toc454362094)

[1.2 Audience 4](#_Toc454362095)

[1.3 Related documents 4](#_Toc454362096)

[2 POINTS TO CHECK IN e\_trad\_ecrans after initial delivery 5](#_Toc454362097)

[3 POINTS TO CHECK IN e\_trad\_ecrans when delivering patch related functionalities 7](#_Toc454362098)

[3.1 Generating a patch with a resource file (.sef file) 7](#_Toc454362099)

[3.2 Generating a patch with multiple .res files 7](#_Toc454362100)

# DOCUMENT DESCRIPTION

## Scope

The purpose of this document is to describe the points to check required to perform the initial delivery of the translation tool to the client server.

## Audience

The intended audience of this document are the members of the following departments, involved in the process:

* PMO
* Communications
* Installer
* CDMT

## Related documents

* dev\_client\_management\_translations\_tech\_draft\_x.yz\_EN.doc
* codix\_clients\_managing\_translations\_internal\_procedures\_x.yz\_EN.doc

# POINTS TO CHECK IN e\_trad\_ecrans after initial delivery

To make sure the translation tool works correctly when installed on the client translation instance, the next verifications have to be performed:

* Ensure the IMX\_FORMS table in e\_trad\_ecrans is delivered. If the list of forms in e\_trad\_ecrans is empty, this means that the corresponding table has not been installed. Responsible: Communications team
* Check the environment variables related to tags, languages and activity are properly set:
* *$FORMS\_LNG –* defines the language
* *$IMX\_FORMSTAG –* defines the client tag
* *$IMX\_FORMSACTIVITY –* defines the client business

Responsible: Communications team

* Check if the following shell scripts are installed:
* *ls -l ${IMX\_HOME}/bin/TranslFromRes.sh – translates and generates the form;* Responsible: Communications team
* *ls -l ${IMX\_HOME}/bin/exportFrm.sh - creates the res files;* Responsible: Communications team
* ls -l *${IMX\_HOME}/*bin/ml\_patch.sh - creates the patch with the translation; Responsible: Installer team
* Check if the following program is present:
* ls -l *${IMX\_HOME}/src/imxtrad –* creates the res files and translates the screen; Responsible: Communications team
* Check the installed *i*MX version:
* If it is an old version, the new one has to be installed and the 64 bit res files have to be deleted, then all screens have to be recompiled. Responsible: Communications team
* Check if e\_trad\_ecrans is set in client mode:

*select value FROM imx\_instance WHERE key = 'IMX\_DEVELOP\_MACHINE';*

*If no results are returned, then the client mode is on.* Responsible: Communications team & then CDMT team during the test.

* Check if the tables below are installed:
* *ML\_RES -* contains unique identifiers of the strings (RID).
* *ML\_FORMS -* gives the relations between screens and strings.
* *ML\_TRAD -* stores stings translations
* *ML\_LANG –* stores translation languages
* *ML\_TAG –* stores client tags
* *IMX\_FORMS –* stores the list of screens used by the client
* *ML\_CONTROL –* stores control types
* *ML\_ACTIVITY –* stores client business activities
* *DCL\_ML\_FORMS   -* keeps history of the screens for which are added translations
* *DCL\_ML\_RES  -* keeps history of the original strings for which are added translations
* *DCL\_ML\_TRAD –* keeps history of the translations

This check is performed with the translation test.

Responsible: Communications team checks if all tables are installed, if not, sends a request to DBA to install them🡪CDMT team makes the test

* Check in MMPI if the buttons surrounded in red are active:



Responsible: Installer team, CDMT

These buttons are used to create a PR containing scripts that will update the client translation instance with Codix STD values at the same time ensuring that no translation done by the Client is lost. The script triggered by ‘**Add trans. update scripts’** button update the translation database in ml\_res, ml\_trad, ml\_forms and ml\_control tables, whereas the script triggered by ‘**Add menu and alerts update scripts**’ update the menus, submenus and the alert messages in their corresponding tables v\_menu, t\_ssmenu and v\_translation tables.

NOTE: When a new instance is created on Codix/or Client side all existing translations in the supported by the client languages from these tables must be delivered to this new server. PMO indicates the server to use where these tables are already translated.

Responsible: PMO, Communications team

Together with the PR, these buttons update the translation DB on the Client DEV machine (Codix side). This is done so before the PR is deployed we verify on DEV that the scripts work correctly.

It is not recommended to create PR-s with res. files created on our server and deliver to client machine because they will override their translations.

Exceptions may occur in some particular cases when a full synchronization of client translation base with ours has been recently done and after a check if the client has not made any subsequent modifications after that.

Res files are created on client translation instance and then the generation of the forms after execution of the scripts above is performed by Codix.

For more details on the cf: *Codix-clients\_managing\_translations\_internal\_procedures\_x.x.x\_EN.doc*

# POINTS TO CHECK IN e\_trad\_ecrans when delivering patch related functionalities

## Generating a patch with a resource file (.sef file)

* Check if in e\_trad\_ecrans the button ‘Prepare for patch’ works correctly.

Responsible: CDMT team together with installers team make a test

This button generates a patch containing the resource file for the selected form and language and allows the users to deliver themselves along the chain

The script triggered by this button is “ml\_patch.sh” and is stored in CVS on REFFR

refbg2 imx-/app/imx/base/bin> cvs log ml\_patch.sh

log ml\_patch.sh

RCS file: /app/cvs/repo/bin/ml\_patch.sh,v

Working file: ml\_patch.sh

head: 1.8

date: 2015/09/08 15:09:05; author: tbratanov

* If the operation ends with success, the output is as follows:

Tue Sep 29 10:47:13 EEST 2015 | -<INFO>- | ##### --> The patch is created ! <-- #####

Checksum: 2152170615

File location: /omen/intra/imx/base/fweb/self\_patches/hatchery/SfPatch20150929\_e\_screen name\_en.res.sef

* In case of error, there will be an error output and a log file in ${IMX\_TRACE}

Once generated the patch has the following format “SfPatchYYYYMMDD\_e\_screen name\_lng.res.sef” and is located in:

${IMX\_HOME}/fweb/self\_patches/hatchery

Where:

S  - Made through the screen

f  - Front end layer

Patch - identification that it is a patch

YYYYMMDD - Year, Month and Date of creation

e\_screen name\_lng.res - resource file name

.sef - Indication that it is a Self extracted file

When the patch is created the sef file should be delivered to the remote instance at the location below:

${IMX\_HOME}/fweb/self\_patches/install/

The installation of the patch is simple, it is enough to execute it like this:

ksh ./SfPatchYYYYMMDD\_e\_trad\_ecrans\_en.res.sef

## Generating a patch with multiple .res files

* Check if the following data base objects are synced from REFBG2

TABLE       CL\_TRANSL  
TABLE       CL\_TRANSL\_PATCH  
INDEX       CL\_TRANSL\_PATCH\_ID\_IDX  
SEQUENCE    CL\_TRANSL\_PATCH\_SEQ\_UN\_ID  
TRIGGER     CL\_TRANSL\_PATCH\_UN\_ID\_TRG  
INDEX       PK\_CL\_TRANSL\_PATCH

Responsible: DBA

* Check if the following data base tables are synced from REFBG2
* V\_TRANS\_SED
* CL\_TRANSL
* DCL\_ML\_TRAD
* ML\_TRAD
* BRREGLEEDITEUR
* DCL\_BRREGLEEDITEUR
* DCL\_F\_PARENR
* DCL\_F\_PARFAC
* DCL\_V\_DOMAINE
* DCL\_V\_ELEMFI
* F\_PARENR
* F\_PARFAC
* T\_SSMENU
* V\_DOMAINE
* V\_ELEMFI
* V\_MENU
* V\_TRANSLATION

Responsible: DBA

* Check if the script *${IMX\_HOME}*/bin/cl\_res\_patch.sh revision is 1.6 or newer
* Create destination install directory: mkdir -p ${IMX\_HOME}/fweb/self\_patches/installer/

Responsible: Installers

* Check e\_trad\_ecrans.fmb is version 1.106 or higher

Responsible: Forms

* Check if in e\_trad\_ecrans the button ‘Prepare for patch’ works correctly.

Responsible: CDMT team together with installers team make a test

This button generates a patch containing the resource files for the selected form and language and allows the users to deliver themselves along the chain.

The script triggered by this button is “cl\_res\_patch.sh” and is stored in CVS on REFFR

refbg2 imx-/app/imx/base/bin> cvs log cl\_res\_patch.sh

RCS file: /app/cvs/repo/bin/cl\_res\_patch.sh,v

Working file: cl\_res\_patch.sh

head: 1.4

date: 2016/05/16 15:06:03; author: tbratanov

* If the operation ends with success, the output is as follows:

*Fri Dec 4 16:42:00 WET 2015 | -<INFO>- | ##### --> The patch is created ! <-- #####  
/imx1/base/fweb/self\_patches/hatchery/CmPatch20151204\_164157.tar.Z  
1470793304 711859 /imx1/base/fweb/self\_patches/hatchery/CmPatch20151204\_164157.tar.Z*

* In case there are no selected resource files the following message pops up:

*Fri Dec 4 16:41:40 WET 2015 | -<INFO>- | ##### --> Nothing to do. <-- #####*

* In case there is an error during generation the log file will be printed, for example with missing res file, the output is as follows:

*Fri Dec 4 16:52:45 WET 2015 | -<ERROR>- | ##### --> Error stop in cp /imx1/base/fweb/e\_affctra\_en.res /imx1/base/fweb/self\_patches/hatchery/CmPatch20151204\_165243/ <-- #####  
The log file is: /imx1/base/tmp/cl\_res\_patch.sh\_20151204165243.log*

* Once *generated the patch has the following format “****CmPatchYYYYMMDD\_HHmmss.tar.Z****” and is located in:*

${IMX\_HOME}/fweb/self\_patches/hatchery

Where:

C – Client generated patch

m – Middle tier

tar.Z – Compressed format

YYYYMMDD\_HHmmss - Year, month, date and time of creation

When the patch is created the file should be delivered to the remote instance at the location below:

${IMX\_HOME}/fweb/self\_patches/install/

The installation of the patch is simple, it is enough to execute the following commands:

cd ${IMX\_HOME}/fweb/self\_patches/install/ 🡪 to go to the installer’s directory

zcat **CmPatchYYYYMMDD\_HHmmss.tar.Z** | tar –xf – 🡪 to de-archive the file

cd **CmPatchYYYYMMDD\_HHmmss** 🡪 to go to the de-archived directory

./Install.sh 🡪 to install and to generate the file on the server

*The PR* [*https://asparuh.codixfr.private/prt/issue/KBCCFDEV-814/pr/995898/*](https://asparuh.codixfr.private/prt/issue/KBCCFDEV-814/pr/995898/) *is created to serve as a template when delivering this functionality to clients. However, only files and tables can be taken directly from there. As for the screen and script please take the* ***latest*** *available version.*