Unity Comments translator.

Contents

Document Revisions:

Version	Date	Author	comment
1.0	25/3/2016	P.Jouvenel	Creation
1.1	10/6/2016	//	Split initial file into 2 separate documents (comments, iocanning)

1. Disclaimer:

These tools are provided as is and may not have been tested exhaustively. The use of the tool does not imply any responsibility for Schneider-Electric.

2. General purpose:

These tools are intended to provide help to fasten action in unity and or do bulk operations.

They are provided primarily as command line tools. For the sake of example it can also provide an Excel add Ins or any other GUI. Excel extensions are provided as a zipped setup.

NB: Excel add Ins require at least excel 2010 installed

3. Trouble shooting:

In some cases the "add in" may not load on excel 2010. This is usually due to a wrong version of the [Microsoft VSTO 2010 Runtime] (vstor_redist.exe) – Minimum required version: 10.0.60724

The folder Pre-Requisite contains the redistribution for:

- .net 4.61
- Microsoft VSTO 2010 runtime

4. License:

This open source project is licensed by Schneider-Electric under MIT agreement.

Please refer to License file for more details

5. Unity Comments extraction:

This tool allows to extract and re-inject translated/modified comments into unity applications. The comments can be translated outside. It allows bulk operations.

4.1. Extracted file format (pivot format)

The extracted informations are stored in xml format. The xml schema and example:

comments.xsd   comments.xml
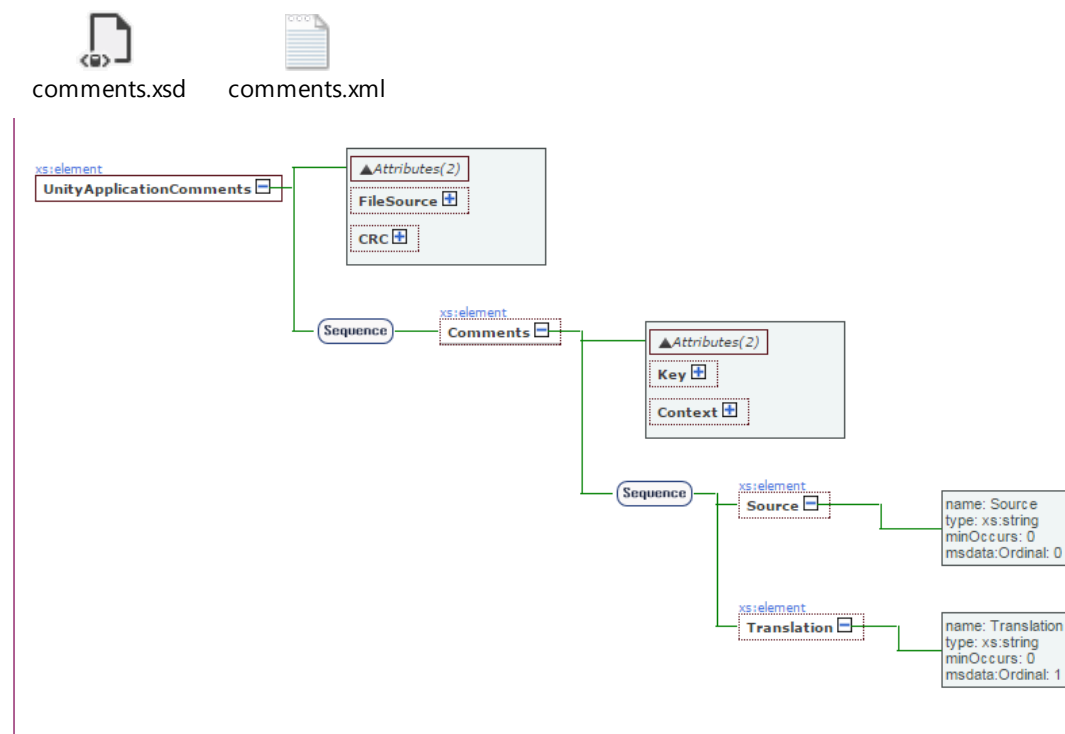
Example:

(Arabic translation)

4.2. Command Line tool:

UnityTranslator.exe {command} {parameter1} {parameter2}

Command	Parameters1:	Parameter 2:	Comments
Extract	IN: Path of the Unity application file to read comments from. Could be STU/STA/ZEF.	OUT : xml file to store the resulting model (comments)	Pre-condition: {parameter1} is a valid unity application file. Reads all comments from application {parameter1} and save it into (parameter2). ⇨ Post-Condition: the file contains all comments from the unity applications.
Translate	IN-OUT :Path of the xml file which contains the comments to translate	IN : Language to translate to	Pre-condition: {parameter 1} is a valid comment xml model. {parameter2} is the code of a supported language. The tool will automatically translates the comments into the given language using Microsoft translate service. NB: a key for Microsoft translate is included but is limited to 2 000 000 char per month. 1. The resulting xml contains all comments and their translation in the given language {parameter2}. Remark: the original xml (parameter1) file is saved as .back before being translated.
Apply	IN :Path of the xml file containing the comments and their translations	OUT: Path of the Unity application file to update comments (.ZEF).	Pre-condition: {parameter1} is a valid file. Updates application provided in {Paramater2} with the translations in the model {Parameter1}. 2. The resulting applications are conform to the model.

4.3. Restriction / usage

The key used to require automatic translations are stored in UnityComments.dll.config.
The tool support up to 4 keys. 2 basics are provided but should be replaced with your own.
To do that:

1) Create a keys in Microsoft translate tool as defined here: (steps1-4)
   https://msdn.microsoft.com/en-us/library/mt146806.aspx
2) Update the config file with `MicrosoftTranslatorAccountkey` as your <u>client ID</u>
   And `MicrosoftTranslatorAccountPwd` as your <u>client secret</u>.

NB: the keys are ordered. The tool will try the first, if it fails the second …. So put your own at first.

UnityComments.dll.config. extracted part:

```
<userSettings>
    <SchneiderElectric.UnityComments.Properties.Settings>
        <setting name="MicrosoftTranslatorAccountkey1" serializeAs="String">
            <value>se-unitytranslator</value>
        </setting>
        <setting name="MicrosoftTranslatorAccountPwd1" serializeAs="String">
            <value>FRWWzrS22b5x0nrhnI2vljYenTbv+LhxMF4SAdKQaF4=</value>
        </setting>
        <setting name="MicrosoftTranslatorAccountkey2" serializeAs="String">
            <value>unitytranslator-se</value>
        </setting>
        <setting name="MicrosoftTranslatorAccountPwd2" serializeAs="String">
            <value>L3McgaNVzU+RhmPN2pK5JlJjaaRGHyBqmPT3Z4REN0M=</value>
        </setting>
        <setting name="MicrosoftTranslatorAccountkey3" serializeAs="String">
            <value />
        </setting>
        <setting name="MicrosoftTranslatorAccountPwd3" serializeAs="String">
            <value />
        </setting>
        <setting name="MicrosoftTranslatorAccountkey4" serializeAs="String">
            <value />
        </setting>
        <setting name="MicrosoftTranslatorAccountPwd4" serializeAs="String">
            <value />
        </setting>
    </SchneiderElectric.UnityComments.Properties.Settings>
</userSettings>
```

4.4. Log

A log file is created and contain all actions/ errors.
It is located in: %programdata%\Schneider Electric\Translator

4.5. Examples:

Extract:

Apply:

Translate:

4.6. Excel add ins:

The Excel add in is designed to help doing bulk modifications/translations to all comments in a unity application. It allows to display, and modify the displayed comments using the power of excel. As it is based on the same xml model (cf 5.1), it may be useful to mix batch operations using command lines and edition operation using excel sheet.

After installation of the unity comment excel plugin, open Excel.

In the add-ins tab, you will find a bar with green buttons:
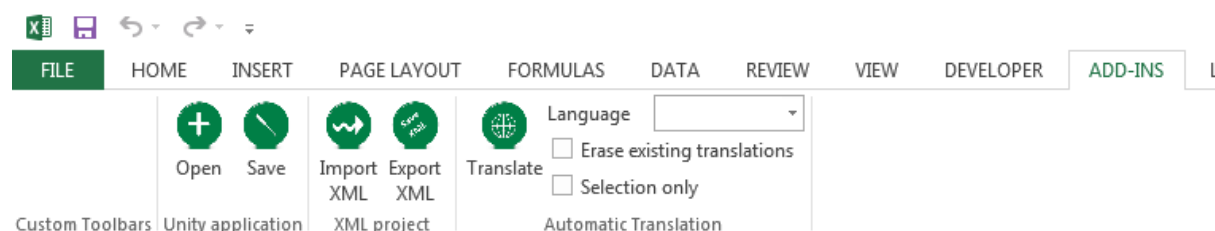
The buttons allow to pilot the extraction translation and generation of the comments.

Usually the flow is:

➢ open unity project.

This will open the project, read the content, extract the comments and display them.
In this result sheet, the model structure is flat. Each line represent one comment in the source application. The second column will contain the translations/modifications of the comment at this

stage it is empty (just read an application), and then in column D the context is displayed. The context helps the operator to understand from where comes the comment: is it a comment in a code section or in the station properties or on a variable etc….

➢ Modify / translate the comments for example copy column A in column B and modify some of them. NB: during the save operation, if a comment is not translated/modified (empty col B) the original comment is used.

➢ Use the automatic translation: the tool allow to use Microsoft translator to propose a first version of the translation for the comments. For sure it needs to be checked but could be a quick first loop.

When using the automatic translation, you need to:
- Select the target language (source language will be determined automatically)
- Adapt the combinable options as desired:
  o Check "erase existing translation" if and only if you want to re translate all comments, even the one already done (non-empty col B). If this option is unselected, only the non Blanc items will be translated.
  o Check "selection only" if you want to submit only the selected items to the translator. If not selected, the whole file is submitted.
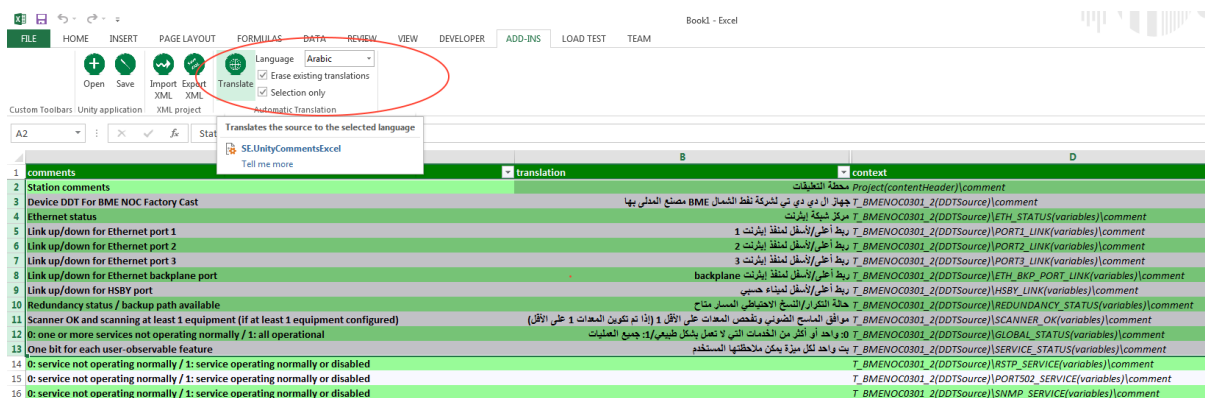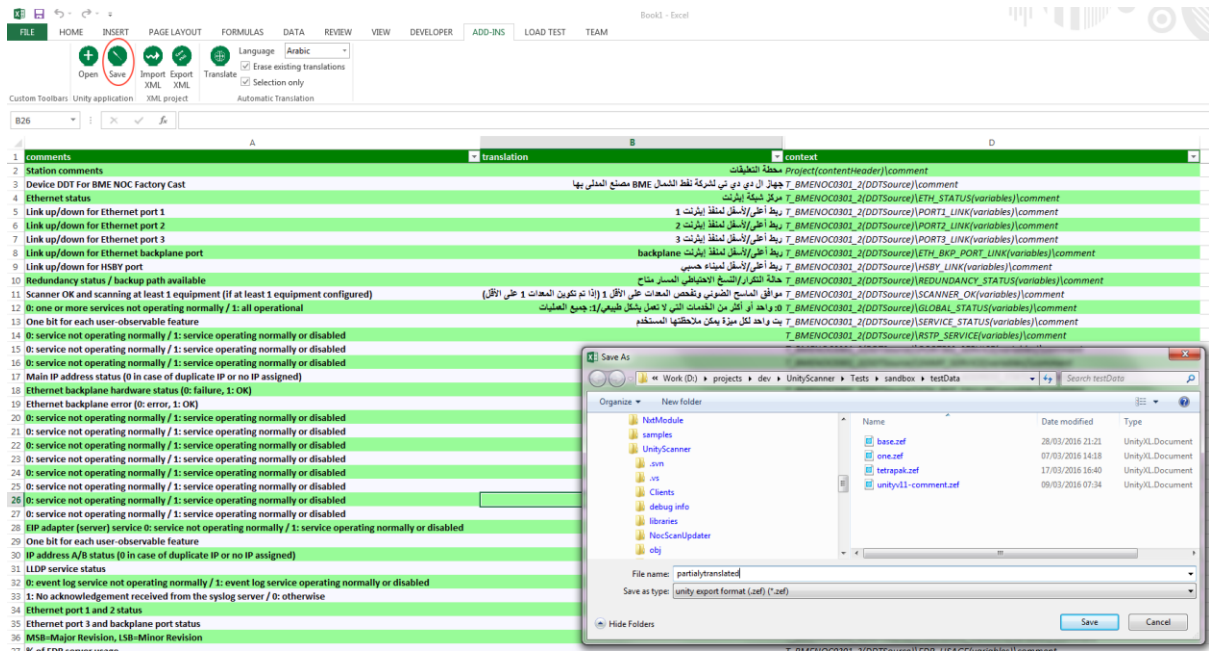

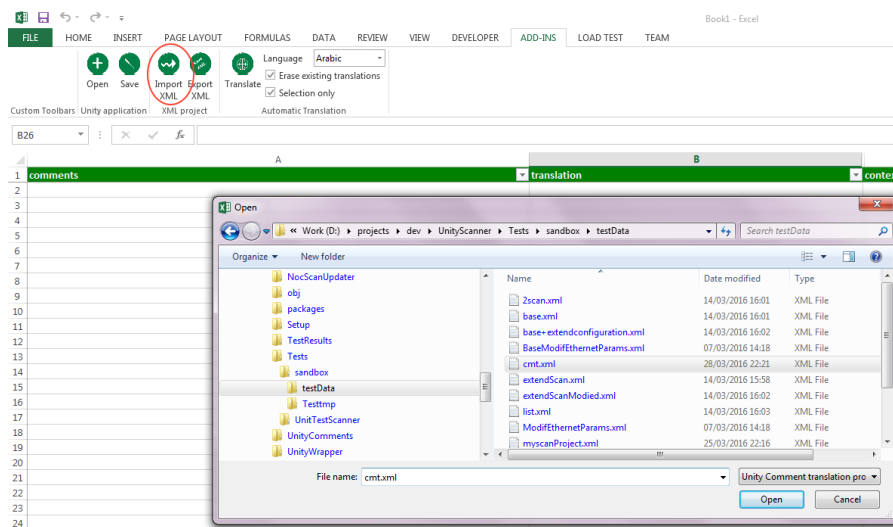➢ Save the model, to be able to reopen it more quickly later.
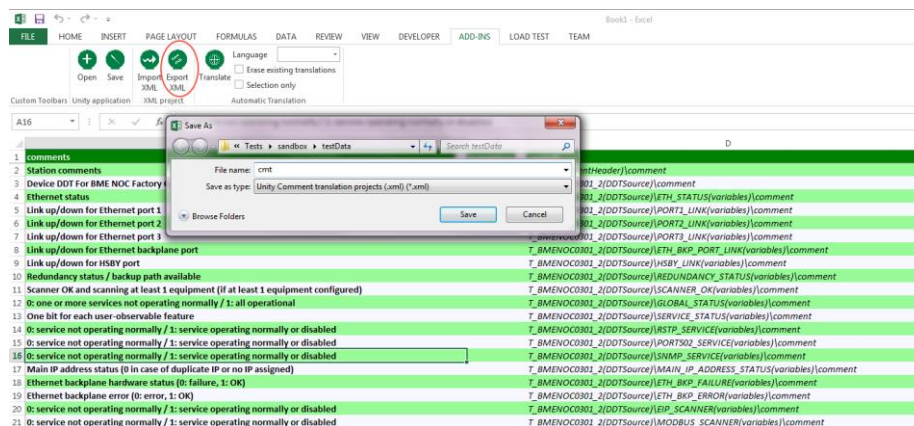
➢ Update unity applications

This will update the application with the translations/modifications.

➢ Load model

This will reload the xml file into the worksheet