OCT – Open Cinéma Trancoder Translation

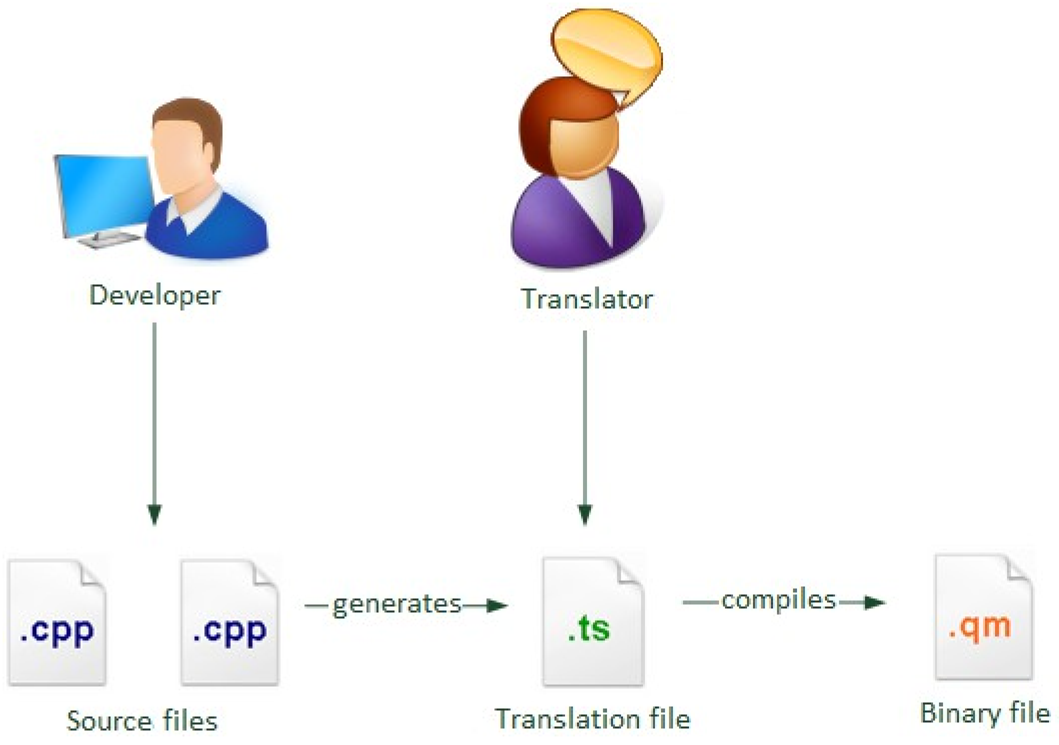
Denis SAUNIER, William LE COROLER,
Thibaud LAMARCHE, Romain BOUTIN
(and for this writing : Geoffrey BERGE)

Referring professor : Philippe MESEURE

catalogue ouvert du cinéma

Summary

Software translation



Development

The first step of the translation is to write the code appropriately so that the translator can retrieve all messages to be translated.

Use QString

Qt uses its own class QString to manipulate strings. This class supports Unicode so all types of characters are allowed. Strings that must be translated in the software program should be handled with a QString.

```
        QString s = "MyString";
```

The tr() method

The tr() method indicates that the string can be translated.

```
        QMessageBox::warning(this,QString(tr("CanBeTranslated")),
Qstring("CanNotTranslated"));
```

You can add a message to explain the context to the translator.

```
        saveAction->setShortcut(QKeySequence(tr("Ctrl+S", "Shortcut for save")));
```

File translation (.ts)

The software program already generates oct_fr.ts file for the French translation because in the oct.pro file, there is the following code:

```
TRANSLATIONS = oct_fr.ts
```

To generate a new translation file, it must be added at the end of the line of code. Example to add a Spanish translation file:

```
TRANSLATIONS = oct_fr.ts oct_es.ts
```

In order to generate the translation file you can use QtCreator or you can use the console.

- In a console, go to the folder of your project.
  Enter:

  ```
  lupdate oct.pro
  ```

- With QtCreator:
  Tools → External → Linguist → Update Translations (lupdate)

A message of this type is obtained:

```
Updating 'oct_XX.ts'...                                            a
       Found XXX source text(s) (XXX new and XXX already existing)
```

It will update only the strings that have changed. The generated file is in the project folder.

Translation with QtLinguist

To begin the translation you must open the oct_XX.ts file with QtLinguist.

Using Ubuntu or Debian, if QtLinguist is not installed, you must install it via a terminal qt4-dev-tools.

Enter:

```
sudo apt-get install  qt4-dev-tools
```

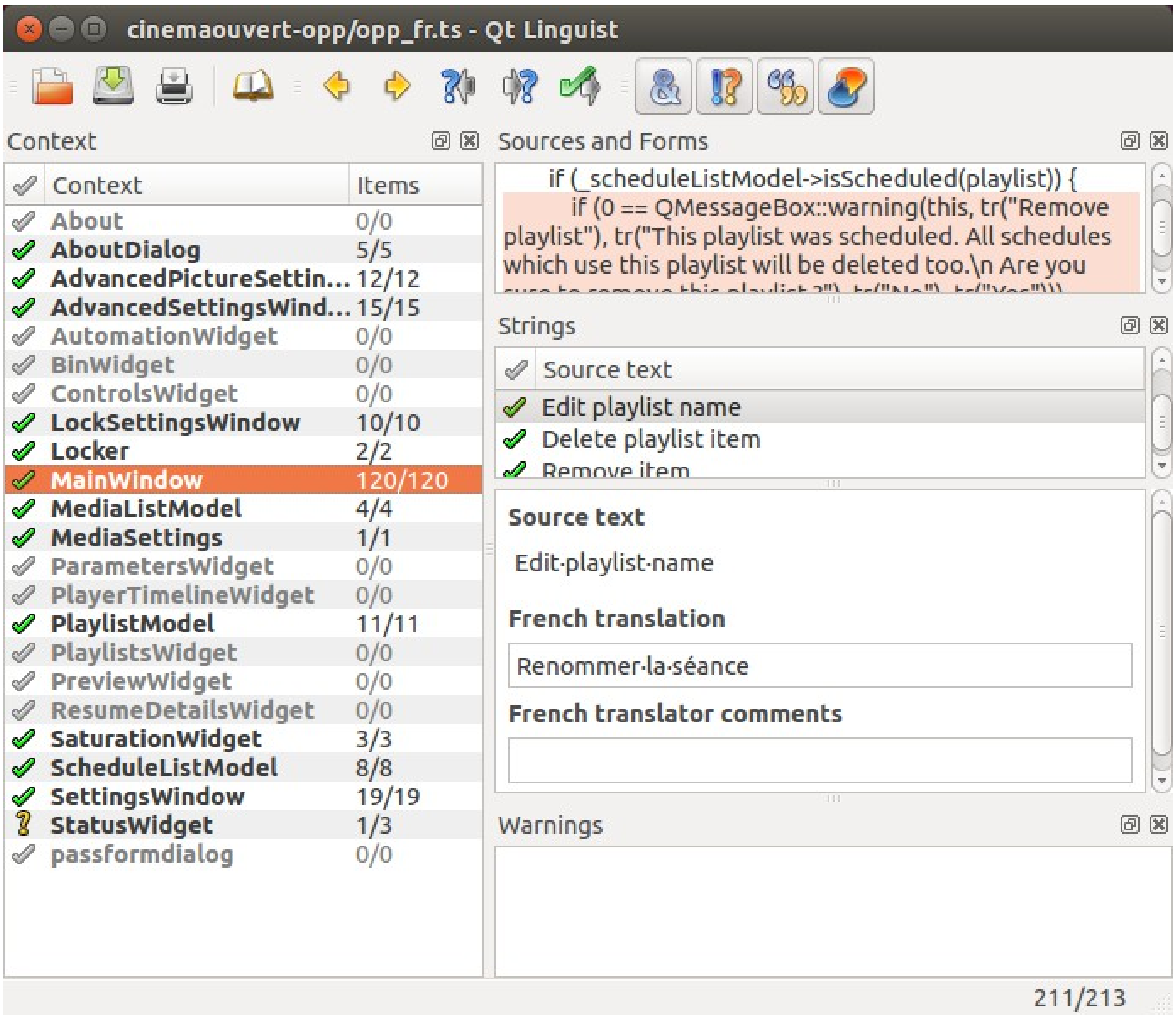
This is the interface of Qt linguist.

You can see that the interface is separated in several windows:

<u>Context:</u> list of source files containing translatable strings.

<u>Sources and forms:</u> contents of the file selected.

<u>Strings:</u> list of translatable strings of the file selected.



✓	Context	Items
✓	About	0/0
✓	AboutDialog	5/5
✓	AdvancedPictureSettin...	12/12
✓	AdvancedSettingsWind...	15/15
✓	AutomationWidget	0/0
✓	BinWidget	0/0
✓	ControlsWidget	0/0
✓	LockSettingsWindow	10/10
✓	Locker	2/2
✓	MainWindow	120/120
✓	MediaListModel	4/4
✓	MediaSettings	1/1
✓	ParametersWidget	0/0
✓	PlayerTimelineWidget	0/0
✓	PlaylistModel	11/11
✓	PlaylistsWidget	0/0
✓	PreviewWidget	0/0
✓	ResumeDetailsWidget	0/0
✓	SaturationWidget	3/3
✓	ScheduleListModel	8/8
✓	SettingsWindow	19/19
?	StatusWidget	1/3
✓	passformdialog	0/0

```
    if (_scheduleListModel->isScheduled(playlist)) {
        if (0 == QMessageBox::warning(this, tr("Remove
playlist"), tr("This playlist was scheduled. All schedules
which use this playlist will be deleted too.\n Are you
```

✓	Source text
✓	Edit playlist name
✓	Delete playlist item
✓	Remove item

Source text

Edit playlist name

French translation

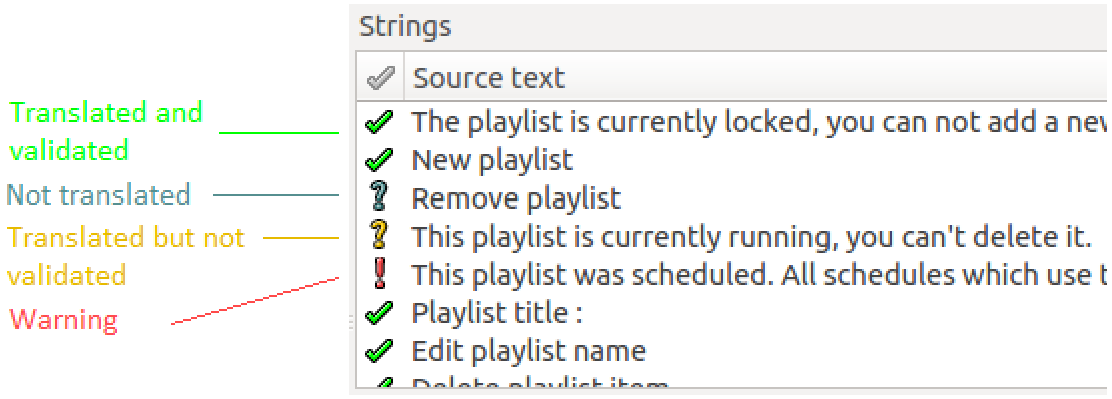
Renommer la séance

French translator comments

Under "Strings": there is the English version of the string, must give the translated version.

Warnings: displays useful warnings such as "Translation does not end with the same punctuation as the source text".

Each message can have 4 states:



✓	Source text
✓	The playlist is currently locked, you can not add a nev
?	New playlist
?	Remove playlist
?	This playlist is currently running, you can't delete it.
!	This playlist was scheduled. All schedules which use t
✓	Playlist title :
✓	Edit playlist name

To validate the translation, you must click on the "? ".

Binary file (.qm)

You should now compile the translated file (.ts) into binary file (.qm).
To generate the binary file you can use QtCreator or you can use the console.

- In a console, go to the folder of your project.
  Enter:
  To compile the selected file:
  ```
  lrelease  oct_XX.ts
  ```
  To compile all the project files:
  ```
  lrelease oct.pro
  ```
- With QtCreator:
  Tools → External → Linguist → Release Translations (lrelease)

A message of this type is obtained:

```
Updating 'oct_XX.qm'...                                    a
        Generated XXX translation(s) (XXX finished and XXX unfinished)
```

You should know that lrelease compiles only strings marked as completed (with the green symbol in QtLinguist).

The generated file is in the project folder.

Load the file in the application

The file is loaded in the constructor() of the OCTDispatcher.cpp file.

```
...

/*Translation file*/
QString translationFile = "oct_";
translationFile+= m_settings.value("language").toString();

QTranslator translator;
translator.load(translationFile);
qApp.installTranslator(&translator);
```

In production, don't forget to place .qm files into a folder named "language" next to the software.