

Creating localization dictionary using Android SDK resources.

When you install recent versions of Embarcadero RAD Studio you also get installed Android Platform SDK (if the respective install option was activated during install). Android Platform SDK installation includes resources with a huge amount of localization files for different target languages. These localization files are placed under **platforms\android-XX\data\res** folder (where XX is Android version). There are plenty of **“values”** sub-folders with different suffixes that relate to target language. By opening **“strings.xml”** file from those folders you can discover that it contains a long list of resource strings that are localized for target language. Reusing this huge localization database in all our applications could help significantly reduce time and costs of projects localization.

TsiLang Dictionary Manager (starting from version 4.0.7) allows importing these localization strings to dictionary and later you can just use **“Auto-translate”** feature from any TsiLang editor to automatically insert them into your projects. Just few easy steps and you can build solid dictionary with many commonly used user interface terms translated.

Let's start with new dictionary file and name it **“android.dic”** (of course you can use your existing dictionary to extend it with new content!). Run **Dictionary Manager** and you will see its main window:

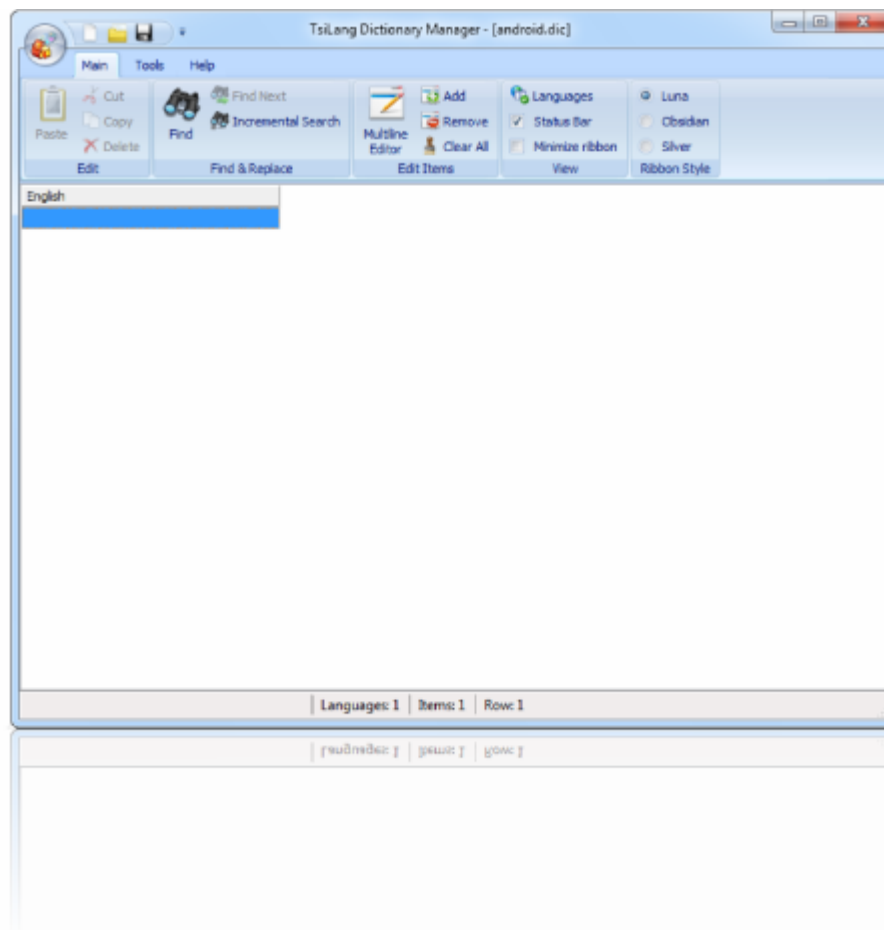


Figure 1. Dictionary Manager Main Window

Select in application's menu **“Import”** command and select Android SDK localization XML file from Platform SDK folder, for example: **“C:\Users\Public\Documents\Embarcadero\Studio\15.0\PlatformSDKs\adt-bundle-windows-x86-20131030\sdk\platforms\android-19\data\res\values\strings.xml”**. Let's will start with content from **“values”** folder (without any suffixes) as it contains default English localization.

In **“Import XML” Wizard** select **“Android SDK Localization File”** option as XML format:

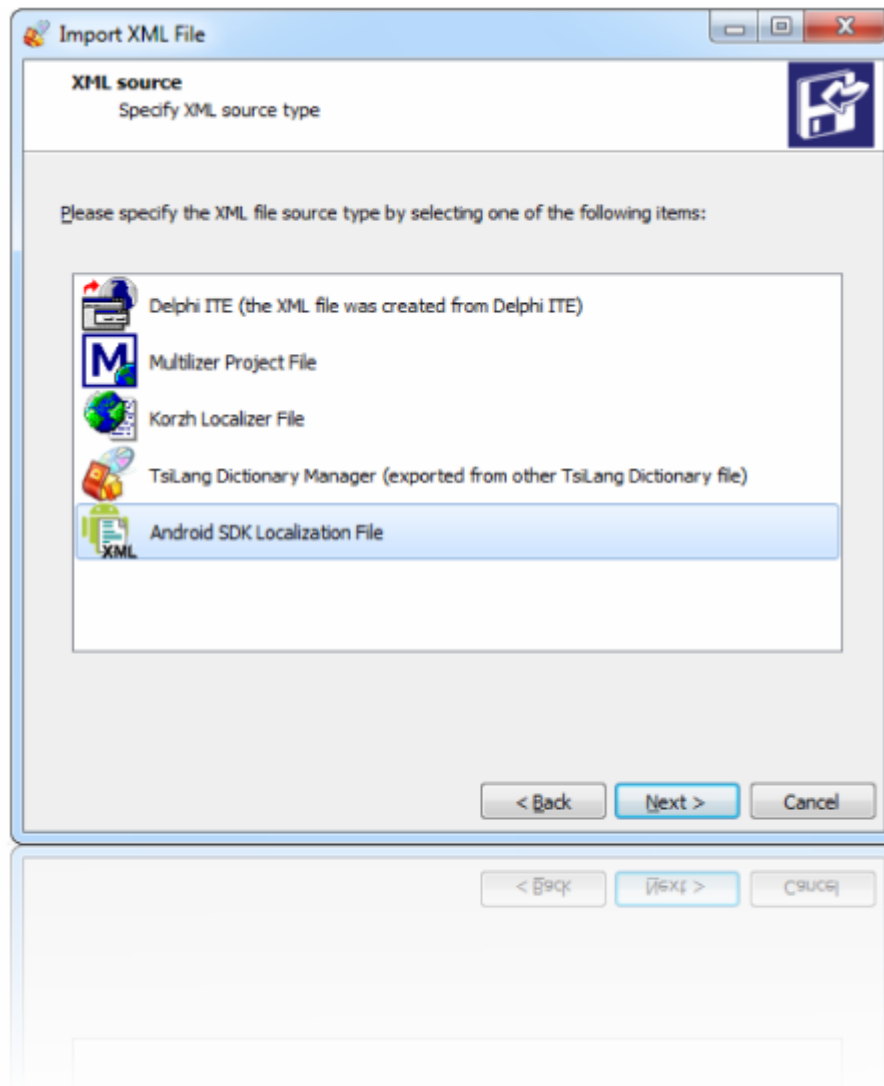


Figure 2. Import XML Wizard

Then click **“Next”** button to continue.

On next step activate **“Simple importing”** option to speed-up importing since our dictionary is empty on this step.

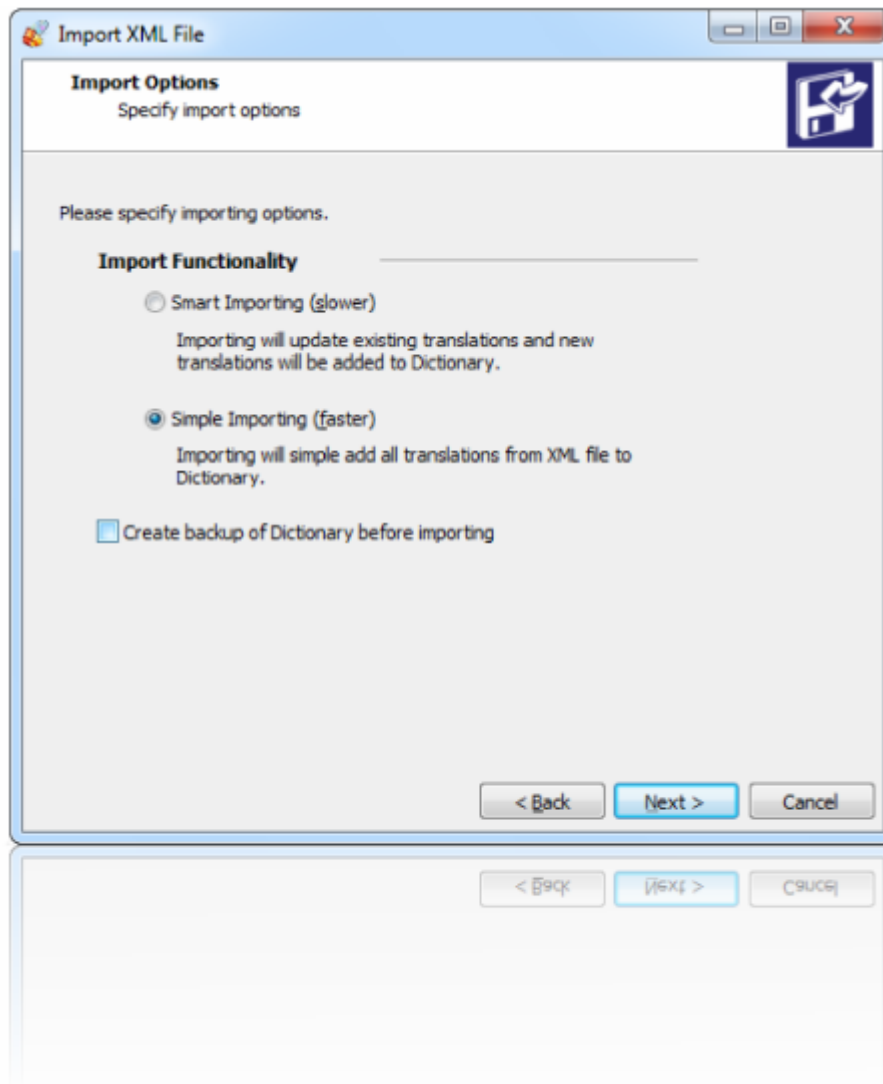


Figure 3. Importing Options

Android Platform SDK localization XML file contains string entries identified with **“name”** attribute and localized content in the value of XML entry. The **“name”** attribute will be also used to identify the entry in the dictionary, so it will be imported as separate language. Select the name for this language in the dictionary, the default value **“Android_Name_ID”** is recommended to use as there is a quite low chance that you have any language with such name. But you can always enter your own name for this language. Also specify the language name for the string localization values from importing file. As we’ve selected **“default”** (English) localization file we will name importing language as **“English”**:

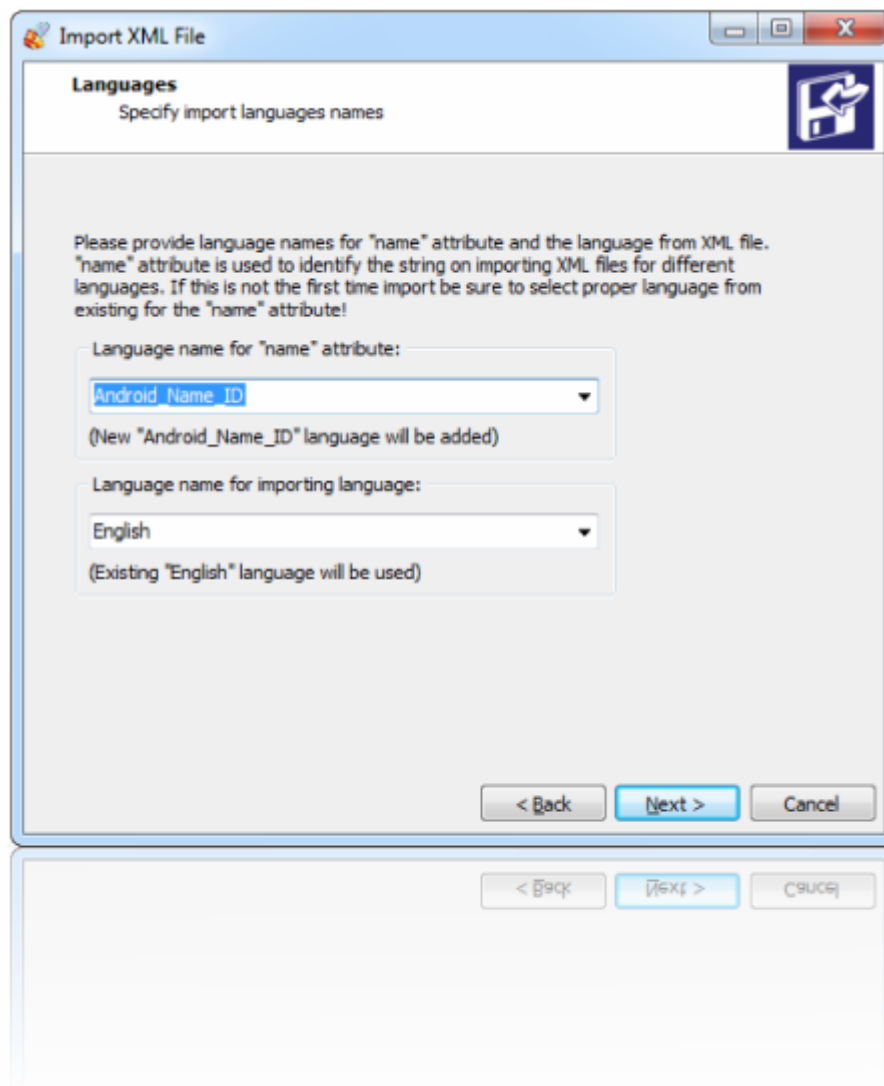


Figure 4. Configuring language names for importing

After you specified language names, you can click **“Next”** and the **Wizard** will perform importing as it had collected all necessary information. After importing is complete, just close the **Wizard** and you will see in the **Dictionary Manager** new languages are imported:

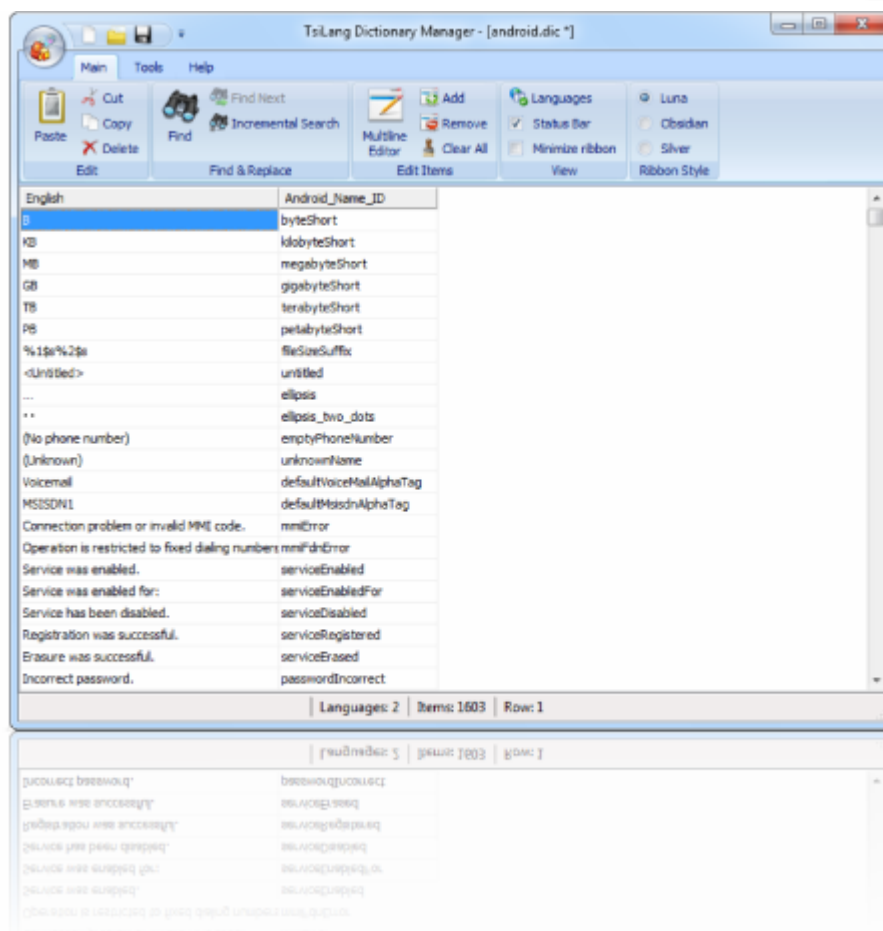


Figure 5. Dictionary with imported English localizations

Now our dictionary includes Android string ID language and English localization for strings used in Android Platform SDK. To add more languages, just run the **“Import XML” Wizard** again with other **“strings.xml”** files from **“values”** sub-folders that correspond to the needed languages. For example, to add **German** language you can import strings from **“values-de”** sub-folder, and to add **French** language you can import strings from **“values-fr”** sub-folder.

Note: just be sure to activate “Smart Import” option on third step of wizard to let it find previously imported corresponding localizations and just update them with localization values from the new language!

After running the **Wizard** for German and French localization files from Android Platform SDK, you will see them appear in the Dictionary Manager:

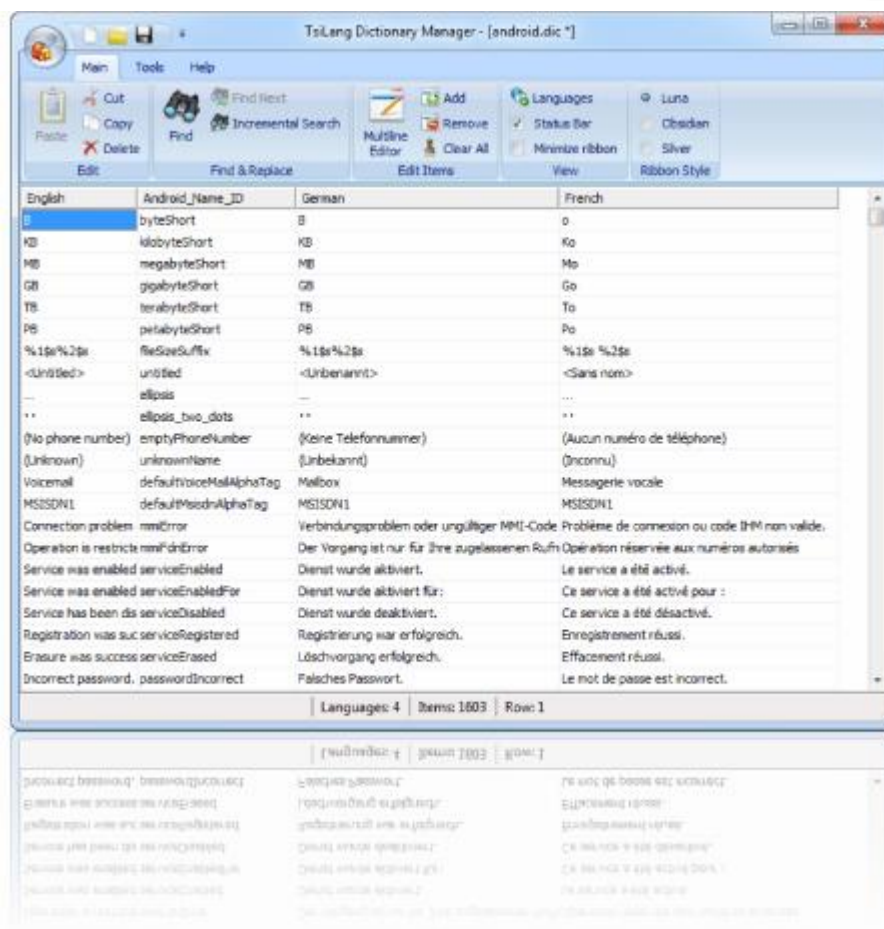


Figure 6. Dictionary with English, German and French localizations

As you can see now, building your own dictionary with more than 1600 localization items required less than couple of minutes. And now you have solid extension to your localization database that might help translate your multi-language applications even without translator service. Just use **“Auto-translate”** feature from **SIL Editor** or **TsiLang Translations Editor** to retrieve translations from dictionary to your projects.

TsiLang Components Suite home page: <http://www.tsilang.com>

Latest version can be downloaded from: <http://www.tsilang.com/download.html>