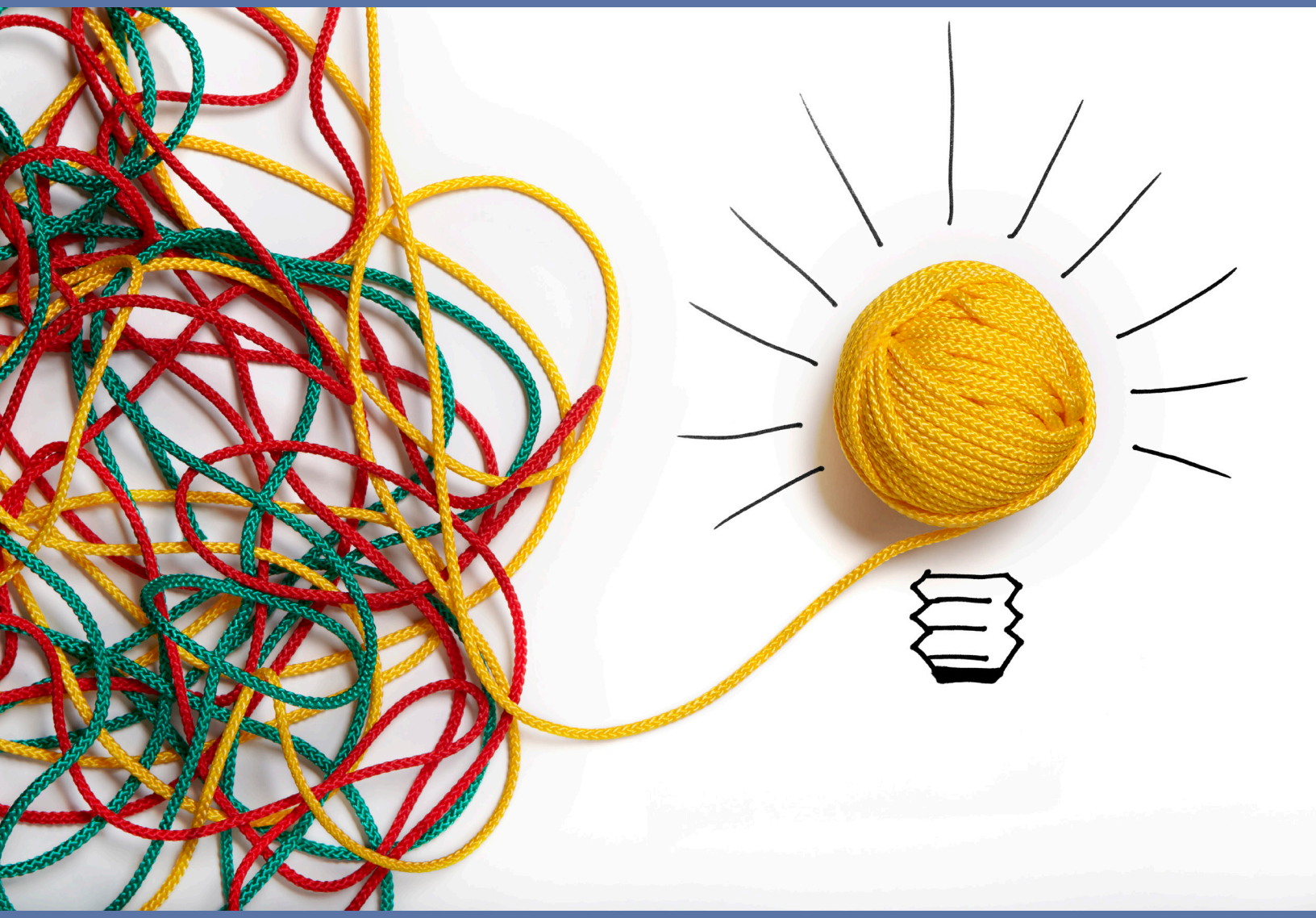

What You Need to Know About Software Translation

(and mobile apps, too)

June 30, 2023

www.idem.com

As you take your product global, you may need to translate the user interface of your software or mobile application. There are several specific things that you can do to:

- keep costs to a minimum,
- hit your target deadline,
- ensure the translation project runs as smoothly as possible, and
- improve the usability of the translated UI.

In what follows, we discuss how to optimize your software translation project on three levels: the big picture of how you send the strings back-and-forth with your translator, the details of how to ensure your English strings are ready to be translated, and the minutiae that can make or break user comprehension of the interface in international markets.

If you're still deciding whether you should even translate your software or app in the first place, check out this related brief.

The Big Picture

Any software or mobile application projects involves several basic steps that set the stage for a successful translation. By understanding and optimizing these steps early, you can ensure an on-budget, on-schedule project.

#1. String Extraction

First you need to get your strings ready. You only want to pay your translators to translate the right strings, of course! Translation providers can work with many different file types (e.g. RESX, XLIFF, XML, PO, CSV, XLSX, etc.), but it is best to confirm your preferred format with your provider and run a pseudo-translation (see orange box) as a precautionary measure. At the end of this pseudo-translation process, you will have files that are ready for translation into any language.

What is pseudo-translation?

Pseudo-translation is an automated process that your translation provider can offer before the translation project even begins. Using the same software tools that the actual translators will employ, pseudo-translation identifies each translatable string and replaces it, either with foreign-looking "gibberish" or with non-linguistic placeholders (such as "$$$$$"). Your team can run a test integration of these strings into your system. Pseudo-translation serves two key purposes:

1. **Ensuring that all strings have been properly captured for translation.** For example, if your team performs a test integration and still finds English words on-screen, these strings have not been properly coded for translation.
2. **Ensuring that no non-strings have been included.** If any part of the actual software code is being captured in the "translatable" content, re-integration will fail. Identifying such an error before translators actually begin working on the strings will save you significant time, particularly if the project is going into multiple languages.

If you have any concerns about font encoding, particularly for accented letters, right-to-left languages, or Asian character sets, the pseudo-translation process can help your team confirm that their planned re-integration processes will work smoothly once they receive the real, final translations.

#2. Context is King

Now that your strings are ready, take a look at them in isolation. If you were not intimately familiar with the product, would you know exactly what each string means? For example:

Sync

It's probably a button, right? So it should be translated as an imperative, as in "Sync this now." But what if, instead, this is actually a label showing the most recent sync date?

Sync: March 1, 2023

This is the kind of information that you want your translators to have on hand while they work. There is no such thing as too much context, so we recommend two steps to maximize the context available to your translators:

1. **Provide comments/descriptions within the code itself.** This gives the translators specific information on your intentions for each string.
2. **Give visual context.** That is, let the translators actually see the interface they are translating. This can be accomplished dynamically (with a testing access and script) or statically (with screenshots taken by your team).

#3. Translating the Strings

Now it's time for your translators to start translating. Ideally, you will work with your provider to complete one language first, making it possible to integrate your processes and theirs (and optimize the shared workflow) while the stakes are still small. This is especially important if you are translating the software or application into many different languages. Caught early, an error in the English file is no big deal. But if your translators discover an error in the English after launching the project into 24 languages, the translation timeline will be substantially delayed: for every language, your provider will need to swap out the new source file and transition over the work that was already completed before the error was discovered.

#4. Verify and Certify

How many times has your team checked the English interface to ensure that the text is displaying properly and everything looks right? Dozens or more! You absolutely want to go through the same process with the translated versions. To save yourself additional effort, you can simply borrow the method you used at the start of the translation process to provide visual context. That is, you can integrate the translation into the testing access (with script) or prepare a set of screenshots that matches your existing English set. Either method gives your translators a one-to-one comparison to make sure that the translations are certifiable against the source.

The Details

At the string level, there are three key elements that can have a dramatic and negative impact on the quality of your translations. In some cases, improper handling of these elements in the English code may even prevent translation altogether. Let's look at them one by one.

#1. Space Limitations

English is a great language for user interfaces because words are typically short and we love to abbreviate things. This can be detrimental to the translation process if there is insufficient space on-screen for the appropriate instructions in a given language. For example, many European languages use fewer abbreviations and their average word length can be 20-35%

longer than the English equivalent. In order to increase the "translatability" of your user interface, plan for a lot of white space in the English version and make buttons much longer than the English word. This will save you from either re-sizing elements in the translations (which will impact your timeline) or asking the translators to create *ad hoc* abbreviations (which will impact user understanding of the interface).

#2. String Concatenation

When you concatenate two strings, you do so to create a grammatically correct English sentence. For example, let's take the following four strings:

Refill
the water
the coupling fluid
reservoir.

These might be concatenated in your system to create the following two sentences:

Refill + the water / the coupling fluid + reservoir.

While some languages will put these words in the same order, some will not. French, for instance, places words that modify a noun after that noun. So the French say "the reservoir (of) water" and "the reservoir (of) coupling fluid." These four strings could therefore not be properly translated into French for this application.

How do you avoid this problem? Simply by ensuring that all strings are complete thoughts. You could accomplish this by creating only the following two strings in English, without any concatenation:

Refill the water reservoir.
Refill the coupling fluid reservoir.

Alternately, you could create a concatenation scenario that separates the elements grammatically into two separate parts, for instance by use of a colon:

A refill is required: + the water reservoir. / the coupling fluid reservoir.

From your translator's perspective, the first option is preferable, but the second option (although non-ideal) still enables the creation of grammatically-correct translations. Your users will likely find the first option easier to understand, as well.

#3. Variables

For the same reason that concatenation can cause grammatical problems for translated interfaces, variables are also discouraged. Many languages are highly inflected, which means that the form of the words changes depending upon number, tense, case, etc. A simple English example of inflection occurs with plurals. For some words, you add an "s" (e.g., one cat, two cats); for some words, you add an "en" (e.g., one ox, two oxen); and for some words, you do nothing at all (e.g., one fish, two fish). Never use a variable for actual text content. For non-content variables, such as reference codes, place them after a grammatical separator like a colon:

An error has occurred. Reference code: %1$d

The Minutiae

Now let's look at some of the seemingly more minor issues that could, in fact, play a significant role in usability. Specifically, these are conventions that differ for markets outside the United States and could therefore lead to user confusion if the interface does not display the appropriate local convention. An increasing number of regulatory standards are recognizing the potential risks associated with medical device user interfaces, including the European MDR (2017/745, paragraph 21.3 in Annex I) and Annex A.2 of ISO/TR 24971:2020, guidance for the application of ISO 14971:2019. Below are some issues you will want to consider in order to ensure usability of the translated software or application.

Date Format

The United States is one of the only markets in the world to use the "mm/dd/yy" format. A European user, for example, would understand 02/05/18 as May 2, 2018 (not February 5).

Time Format

In most parts of the world, the 24-hour clock is more commonly used than the 12-hour clock with its "am/pm" designations. There is therefore a risk of an international user misreading "3:52 pm" as "3:52 am" if they were to read the number and not the text that follows. For the afternoon time, they would expect "15:52."

Decimals and Thousands

The decimal point, used in the United States, is relatively uncommon around the world; the decimal comma is in more widespread usage. By the same token, different countries use differing conventions for separating out the thousands marker…including the use of a period in some markets. In combination, these two conventions can cause a serious risk for user misunderstanding of numerical data. For example, a French user would understand "10,000" as the number ten displayed at three decimals (i.e., 10.000) and would understand 2.379 as a number greater than two thousand (i.e., 2,379).

Units of Measurement

Only a few countries outside the United States make use of the U.S. customary system of measurement, which includes inches, feet, ounces, pounds, teaspoons, and degrees Fahrenheit. Even though these measurement words are directly translatable, most users would find it confusing to use them without performing a conversion to the metric equivalent.

Punctuation

There are only a few example of punctuation differences in non-U.S. markets that could lead to possible user confusion, such as the decimals and thousands separator differences described above. It is worth noting, however, that punctuation standards do differ around the world and it is advisable to ensure your translators can adjust punctuation in the translated user interface for a professional appearance.

Although the text of a user interface is often straightforward, there are several pitfalls that can affect a translation endeavor. However, with a bit of advance planning – both in the English build and at the launch of the actual translation project – you can translate your software or mobile app into any language with relative ease.

About Idem Translations

Founded in 1983, Idem Translations, Inc. is a full-service provider of translation and localization services. Idem specializes in certified translations for medical device, biomedical, and pharmaceutical companies, as well as other organizations and entities working in the life sciences sector, such as contract research organizations, healthcare research centers, and institutional review boards. The company is a WBENC-certified woman-owned business and holds certifications to ISO 9001:2015, ISO 13485:2016, and ISO 17100:2015.

Get Help

For more information about how we can take the risk out of translations for you and your team, please visit us online:

WEBSITE
www.idem.com

TWITTER
twitter.com/IdemTransInc

LINKEDIN
www.linkedin.com/company/143474