

# The Seven Laws of Successful Software Localization

By Frank Dietz

<http://www.frankdietz.com>

## *1. Make localization an integral part of the development process*

Unfortunately, localization is sometimes only an afterthought, and this can lead to all kinds of problems, from cost overruns to the necessity of rewriting code. Therefore, you as a localization consultant should be involved in the project – even in its early stages when crucial decisions such as interface design are made.

## *2. Support those foreign characters*

Accents, umlauts, tildes – foreign languages use special characters that are not present in the English alphabet, and non-Latin alphabets pose different problems altogether. Make sure that your fonts support all the special characters of the language you translate into. A good resource for multi-language font support issues is the home page of the Unicode consortium: <http://www.unicode.org/>.

## *3. Avoid grammatical pitfalls*

We know that the grammatical structures of English can not simply be retained in a foreign language. English sentence structures lend themselves more to a "building block approach". The most important rule is - and you should point that out to your client well in advance – that programmers should consult a translator before hardcoding sentence structures in text strings.

## *4. Give foreign words space to breathe*

You as a translator know how important it is to leave extra space for translated terms, particularly for button labels and menu items. The word "map", for instance, would become "Landkarte" in German. Therefore remind your client to either allow for sufficient extra space or to make these interface elements dynamically resizable.

## *5. Keep terminology consistent*

Large localization projects might be given not just to you, but to a group of translators. Make certain that your project manager provides each of you with a common glossary of terms in order to ensure consistent terminology throughout the project. This is particularly important if the Web site and the documentation are translated by more than one person. It is neither in your nor in your client's interest to confuse the end user by inconsistent naming of menus or screens. If your project is very large, you might even consider a glossary management program or recommend your project manager to provide you with one.

## *6. Have a test plan*

The foreign versions of a program should be tested by native speakers of that language, whether through you, a freelancer who comes on-site, or a specialized testing company. The testing should be done on computers running the appropriate foreign version of the operating system and using foreign-language keyboards.

## *7. Be culturally sensitive*

This is a crucial and very wide-ranging issue, covering everything from paper formats and the structures of phone numbers to ethnic stereotyping and issues of sex and violence. You might not just need to translate a program, but may also have to adapt certain parts of it to conform to different phone systems, measurement units or legal traditions. Even such seemingly "universal" elements as icons might be culture-bound (will, for example, users in your target culture recognize an American mailbox, and if so, what will they associate it with?). When I worked as an in-house translator for an Austin-based computer game company, we had to "tone down" the graphic violence in certain products for the German market, as the legal guidelines there are much stricter in this aspect.

Software localization can be tricky, but also very rewarding. These guidelines will not guarantee the success of the program in foreign markets, of course, but at least they should help you to steer clear of common pitfalls. If you need further information, consider one of the following online resources:

**Silicon Valley Localization Forum:** <http://www.tgpconsulting.com/>

**Localisation Industry Standards Association:** <http://www.lisa.unige.ch/>