

Translations of free software into Irish

Kevin P. Scannell
Department of Mathematics and Computer Science
Saint Louis University
St. Louis, Missouri, USA 63017
`scannell@slu.edu`

Séamus Ó Ciardhuáin
Department of Information Technology
Limerick Institute of Technology
Moylish Park
Limerick, Ireland
`Seamus.OCiardhuain@lit.ie`

October 23, 2006

1 Free Software

An Irish language version of Windows XP was launched by Foras na Gaeilge and Microsoft Ireland in June of 2005 to great fanfare. Given that Microsoft controls about 95% of the desktop computer market, this was clearly a major step forward in the provision of technology to Irish speakers in a native language context. At the same time, tucked away among the recalcitrant 5% of non-Windows users, there is a small community of volunteer translators and software developers that has been enjoying a completely free Irish language desktop system since 2002. This system is based on Linux, a free alternative to the Windows operating system, and includes a complete range of end-user applications such as web browsers, email handlers, office software, and games.

Here “free” has a technical definition¹ which means roughly that the software in question can be copied, modified, redistributed, or even sold by anyone, as long as the redistributed versions preserve these same freedoms for others. While there is no requirement that the software be distributed at no cost, in practice it almost always is². One occasionally hears reference to “open source” software, which, for the purposes of this paper, amounts to the same thing, despite endless hair-splitting in the free software community.

¹See <http://www.gnu.org/philosophy/free-sw.html>.

²Richard Stallman, founder of the free software movement, has described it as free as in “free speech”, not as in “free beer”. Unlike most Romance languages (cf. Fr. *libre* vs. *gratuit*) Irish has a similar ambiguity with the word *saor*, as in *sairse* vs. *saor in aisce*.

A great deal has been written in software engineering circles about the advantages of open source development over traditional proprietary models in terms of software quality. Indeed, free software is now a vital part of the Internet infrastructure in the form of the Linux operating system and Apache web server, and is making substantial inroads in desktop software such as the Firefox browser which has won significant market share because of its features and security benefits. In this paper we will restrict our attention to a discussion of the ways in which free software is particularly well-suited to minority language localisation, directing the interested reader to [2] or [6] for a more general discussion.

Our primary claim in the present paper is that free software offers the only cost-effective and sustainable way to provide a fully localised computer system for minority languages and other languages for which there is limited commercial interest in localisation. There is strong empirical evidence supporting this claim; for example, translations of the core KDE system exist for at least 80 languages³, more than twice as many as are available for Windows XP, with minority languages (Breton, Kashubian, Low Saxon) and under-resourced languages (Kinyarwanda, Kurdish, Xhosa), making up the difference.

One clear reason for this are the costs that must be borne by the end-user – in countries where proprietary commercial products are prohibitively expensive, free software is the only realistic way to ensure that users will have access to these technologies.

A more fundamental reason for the success of free software localisation is that the translations themselves are available under the the same licensing terms as the software, which means that the translations are owned by the community as a whole (to the extent that they are owned at all), and not by Apple or Microsoft. This ensures that as new versions of the software are released, the translations will be updated as needed, even if the original translators have moved on to greener pastures⁴. It also means that all translations can be reused in other free software projects, which greatly accelerates the localisation process and improves quality and consistency across applications. We discuss our translation compendium further in §3 below.

Dependence on state-funded translation and localisation of proprietary software is a dangerous strategy for small languages. Changing economic or political circumstances can cause a government to cut funding. Moreover, there are many languages which will not receive support because of government indifference or hostility. Similarly, it is dangerous to depend on the goodwill of for-profit companies; changing corporate agendas can result in a product being dropped. Many readers will recall the Irish language version of Mac OS that was translated by Everson Gunn Teoranta in the early 1990's. The latest versions of Mac OS are no longer available in Irish because Apple chose not to support this localisation in the long term.

The wider geographic distribution of the user base for free software leads to

³See <http://i18n.kde.org/stats/gui/trunk/essential.php>.

⁴As recently happened with the Scottish Gaelic localisation of OpenOffice.org; the company that localised version 1.0 abandoned the project when their grant money ran out; a group of volunteers is now updating the translation for version 2.0.

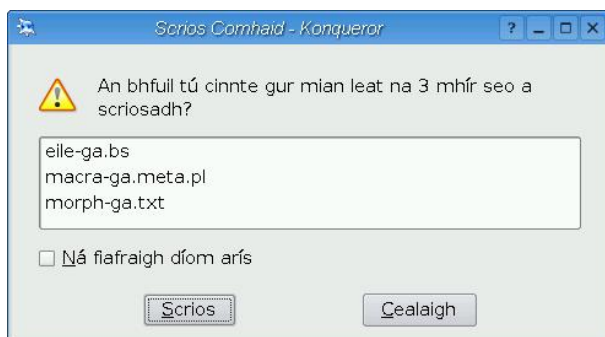


Figure 1: Plural handling in KDE.

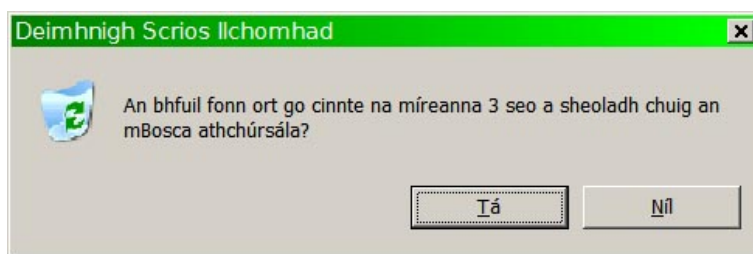


Figure 2: Plural handling in Windows XP.

a stronger emphasis on internationalisation in the development process, which leads to higher quality localisations. For example, most free software is localised using a particular system and file format⁵ that permits sophisticated plural handling, and which can be infinitely configured for non-English languages. As a consequence, where one sees things like “*An bhfuil fonn ort go cinnte na míreanna 3 seo a scríosadh?*” in the Irish version of Windows XP, the same string appears as “*An bhfuil tú cinnte gur mhaith leat na 3 mhír seo a scríosadh?*” in our free system, with correct word order, use of the definite article, and eclipsis or lenition as appropriate on the qualified noun. See Figures 1 and 2.

For the complex technical translations typical of software localisation, the results of an open source approach can be better than proprietary efforts because the translators are often domain experts rather than professional translators. In the case of Irish, for example, many of those working on the free software translations have high levels of expertise in information technology and are able to identify fine points of translation. Although a team might have access to the developers of the software while translating, it is better for both accuracy and speed if the translators understand the technical issues directly. A further benefit is that new terminology can be created quickly and accurately by such

⁵The `gettext` system and PO (portable object) file format; see <http://www.gnu.org/software/gettext/manual/>.

expert translators when needed.

A final advantage worth noting is the shorter development cycle one usually finds in free software projects. When users report errors in one of our localised products, generally speaking we are able to verify and fix the problem within a day or two, and offer a corrected version that can be downloaded and installed if desired. Occasionally the person finding the problem is able to offer a fix, since the source code is freely available. This is in stark contrast to the long wait Windows users face between releases (XP was released in 2001, and its successor, Windows Vista, with promised improvements to the Irish language support, is set to be released in 2007).

This is an all-volunteer effort, which is remarkable given the scope of the project:

- KDE (The K Desktop Environment, which includes the Konqueror web browser and the KOffice office suite, and a wide range of other applications like games, educational software, multimedia, and software development tools). 750,000 words. <http://www.kde.org/>.
- OpenOffice.org (full office suite). 540,000 words. <http://www.openoffice.org/>.
- GNU Translation Project (an assortment of standard Linux applications and utilities). 240,000 words. <http://www.iro.umontreal.ca/translation/>.
- Mozilla (Firefox web browser, Thunderbird email handler, Sunbird calendar). 45,000 words. <http://www.mozilla.org/>.

At over 1.5 million words and growing⁶, this may be the largest Irish language translation initiative since the flurry of novels published in translation by An Gúm in the 1930's.

The present authors manage the technical aspects of the project and also do much of the translation. A core of about six others (thanked below) have provided consistent help with translating. We have a number of additional volunteers who proofread translations, test the software, and provide technical assistance of various kinds (building software, debugging, deciphering opaque strings). Users of the software in the community at large have submitted bug reports, suggestions for improvements, as well as icons and screenshots for publicity. The authors are grateful to everyone who has volunteered his or her free time to this project, most notably: Pat Folan, Marion Gunn, Sean Kelley, Brian King, Iarla Mac Aodha Bhuí, Enda McGuinness, Alastair McKinstry, Brian Ó Broin, Seanán Ó Coistín, and Peadar Ó Guilín.

⁶Our Welsh counterpart Kevin Donnelly has likened translating KDE to painting the Forth Bridge, in that new applications are added to the system at a rate faster than they can be translated.

```

firefox.po + (~) - VIM - Blaosc - Konsol

#: askThirdPartyAfter.label
msgid "about each site I visit"
msgstr "maidir le gach suíomh ar a thugaim cuairt"

#: passwords.label
msgid "Passwords"
msgstr "Focail Fhaire"

#: useMasterPassword.label useMasterPassword.accesskey
msgid "&Use a master password"
msgstr "Úsáid &príomhfocal faire"

#: showPasswords.label showPasswords.accesskey
firefox.po [+] 4,50 Barr
Focal anaithnid

4: #: askThirdPartyAfter.label
  msgid "about each site I visit"
  msgstr "maidir le gach suíomh ar <b>a thugaim</b> cuairt"
Earráid: Urú ar iarraidh

/tmp/v380891/6 [+] 14,9 17%

```

Figure 3: Detection of grammatical errors with *An Gramadóir*. The top half of the screen contains the translations, and errors are reported on the bottom half (here “*Urú ar iarraidh*” indicates “Missing eclipsis” on the verb *thugaim* following the indirect relative particle).

2 Translators’ Toolkit

Ensuring the quality of the translated software has been a top priority since the project began. The first author has developed a suite of tools that can be used to verify translations in a variety of ways.

The heart of the toolkit is a free Irish language grammar checker called *An Gramadóir*, which combines a large lexical database and robust part-of-speech tagging to detect common errors in spelling, initial mutations, and word usage. There are currently more than 3500 rules implemented in the system⁷. See Figure 3.

Translations are reused quite frequently when translating software. This is true both within single applications and between applications in the same domain⁸. Therefore, the use of translation memory software is critical for maintaining consistency across applications and in terms of the efficiency of the translation process (see §3 below). The translators’ toolkit contains a script that performs a “self-consistency” check of the full translation memory by finding discrepancies among existing translations of the same or similar strings. This

⁷The grammar checker is available from <http://borel.slu.edu/gramadoir/foirm.html>.

⁸At last count, Irish localisations were available for at least nine different free text editors: AbiWord, Kate, KEdit, KWrite, leafpad, nano, OpenOffice.org writer, vim, and yudit. Once one or two of these were complete, translating the others became a simple matter.

is useful in cases where we have decided to change the standard translation of a given term and need to ensure that the new choice is reflected throughout all applications.

In the next section, we mention several applications of large translation memories (and, more generally, “parallel corpora”) to the field of natural language processing. Many of these applications rely upon, or are improved by, alignment between translated strings at the sub-sentence level (clauses, noun phrases, or even words). As a result, this kind of alignment is an active area of research [4],[5]. A simple application of our work in this area is the ability to detect gaps (or extra phrases) in translated strings. Errors of this kind are remarkably common when translation memories are used and translators are working quickly. For example, if a translation exists for a string like “Use this button to manage your security devices”, and then in a later version of the same software (Firefox in this case), a slightly different string appears, e.g. “Use this button to manage your security devices, such as smart cards”, there is a chance that an overworked translator might not notice the added phrase and will use the old translation unmodified. When the translators’ toolkit attempts to align these strings, it will fail and warn the translator appropriately.

Computing terminology for Irish has been well-standardised in recent years thanks to the work of Fiontar at Dublin City University and An Coiste Téarmaíochta, and the publication of large terminological databases on sites such as <http://www.acmhainn.ie/> and more recently <http://www.focal.ie/>. Nevertheless, not every translator will immediately recall the preferred term for “token ring” or “proxy server” and occasionally inconsistencies creep in. The toolkit corrects such inconsistencies by checking each translation against a database of known incorrect or deprecated translations. This is done with simple pattern-matching; e.g. if a source string contains the term “proxy server”, a warning is given if the corresponding target string contains the incorrect translation *ionadaí* (or any of its morphological variants).

Finally, many errors in software translation come from a small number of ambiguous English words that translate differently depending on the context, especially when the translator is not technically-oriented or is not familiar with the particular application domain. A common example when translating into Irish is “directory”, which standardly translates to either *comhadlann* (for a file directory) or *eolaire* (for a directory with contact information). We have trained a statistical classifier which is able to guess automatically the intended context of an ambiguous term by examining nearby strings in the file to be translated. For example, when the acronym “LDAP” (Lightweight Directory Access Protocol) appears near the word “directory”, the classifier correctly recognises that this is the *eolaire* sense of the word, and will report an error if *comhadlann* is used in the translation.

3 Parallel Corpora

A parallel corpus is a database consisting of original texts together with their translations into one or more languages. In most cases, the texts and their translations are *aligned*, usually at the level of sentences. Parallel corpora have a number of important applications in natural language processing; they are, for example, the main tool used in training statistically-based machine translation systems. There are also applications to cross-language information retrieval and bilingual terminology extraction (e.g. for lexicography) [5].

In 2003, the first author began the development of the *Corpas Comhthreomhar Gaeilge-Béarla* (CCGB), a large Irish-English parallel corpus, primarily for the purpose of training a statistical machine translation system. When a parallel corpus contains aligned texts from a global language and a minority language, it is also possible to bootstrap linguistic resources for the minority language. We have, for example, used the CCGB to train an Irish “standardiser” that allows pre-standard or dialect texts to be converted automatically to a standard form for indexing and information retrieval purposes [3]. The standardiser has been used to index all of the pre-standard Irish language material on the web for the search engine www.aimsigh.com.

Parallel corpora are closely related to translation memories. Running texts and their translations from a translation memory system can be aligned sentence-by-sentence using well-known algorithms⁹, and can then be incorporated into a parallel corpus. We have done this with a large number of English-Irish text pairs, including literary texts, translations of legal documents, governmental publications and press releases, the Acts of the Oireachtas (see www.achtanna.ie), and the Bible.

In fact, we have taken a much more expansive view than is usual about what ought to be included in a parallel corpus. We have, for instance, included software translations in the CCGB despite the fact that these often come in units smaller than sentences, either single words (“File”, “Edit”, etc.) or short fragments (“Not enough memory”, “Error while reading the database”) not requiring alignment. We have incorporated a large number of Irish-English dictionaries and terminology lists into the corpus as well. In all, the current version of the CCGB contains more than 15 million words in almost 600,000 aligned text segments.

One important aspect of the CCGB is that it is continually growing. As the present authors and their collaborators produce new translations of free software, these are aligned and added to the corpus. In addition, we have scanned a number of Irish language texts for which the English source texts are freely available in electronic form¹⁰. Finally, we also have a web crawler that harvests new English-Irish document pairs from the web, aligns them, and adds them to the corpus automatically [3].

From the point of view of this paper, the real importance of the CCGB stems from the fact that it can be exported in TMX (Translation Memory eXchange)

⁹Such as the *Gale-Church algorithm*; see [1].

¹⁰e.g. from Project Gutenberg, <http://www.gutenberg.org/>

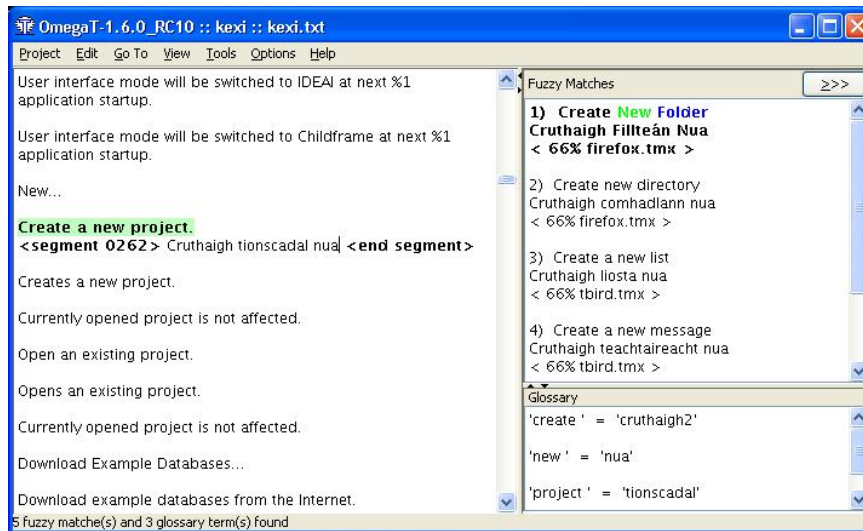


Figure 4: A subset of the CCGB used as a translation memory and glossary with OmegaT.

format; this is an open-standard XML format for the exchange of translation memory databases across platforms and between translation packages. In this way, it is possible to use the CCGB directly with free computer-aided translation software such as OmegaT (or, if one prefers, proprietary software like Trados or Wordfast). See Figure 4.

The main drawback of these standalone translator’s applications is that they do not scale well to handle translation memories on the scale of the CCGB¹¹. The first author is currently developing a web-based translator’s application that uses the CCGB as its back-end, stored on a central server. It will be used in much the same way as the standalone applications noted above, but with the following advantages:

- Access is provided to an up-to-the-minute version of the ever-growing CCGB database.
- The most useful existing translations are displayed using a powerful fuzzy matching algorithm.
- Because the corpus resides on a central server, it does not consume memory or slow down client computers.

¹¹We have found, for example, that OmegaT can be very slow when loading large translation memories, and in fact will crash when fed more than 50,000 segments, a mere fraction of the full CCGB.

- The quality assurance toolkit described in §2 can be applied to submitted translations.
- Use of the software is free, with the condition that submitted translations are added to the central database, and made available to other users making queries.

In the meantime, queries to the CCGB can be made at the following web site: <http://borel.slu.edu/corpas/>. Translators interested in contributing to the effort described in this paper, either by helping translate free software, or by contributing their work to the CCGB, are encouraged to contact the authors.

References

- [1] W. Gale and K. W. Church. A Program for Aligning Sentences in Bilingual Corpora. *Computational Linguistics*, 19(1):75–102, 1993.
- [2] Eric S. Raymond. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. O’Reilly, Sebastapol, 2001.
- [3] K. P. Scannell. Applications of parallel corpora to the development of monolingual language technologies. Preprint, <http://borel.slu.edu/pub/ccgb.pdf>, 2005.
- [4] J. Tiedemann. *Recycling Translations: Extraction of Lexical Data from Parallel Corpora and Their Application in Natural Language Processing*. Uppsala Universitet, Uppsala, 2003.
- [5] Jean Veronis, editor. *Parallel Text Processing: Alignment and Use of Translation Corpora*. Springer-Verlag, New York-Berlin-Heidelberg, 2000.
- [6] Sam Williams. *Free as in Freedom: Richard Stallman’s Crusade for Free Software*. O’Reilly, Sebastapol, 2002.