# Bytes and PCs

*Double-byte languages pose unique engineering challenges not commonly found in single-byte character sets.*

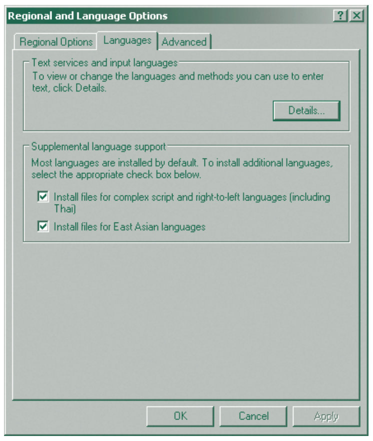
*This adds to the challenge of localising Asian sites, says* AITOR MEDRANO

Aitor Medrano

In the global market, while the localisation industry strives to make e-content accessible to all cultures by overcoming the language barrier, there is still an underlying issue that divides the world into East and West, namely codepages, glyphs, characters and bytes, the mainstay of internationalisation engineers.

East Asian languages such as Japanese, Chinese and Korean are double-byte character set (DBCS) languages. These languages require two bytes to represent a single character, as opposed to one byte for European languages. As a result, double-byte languages pose unique engineering challenges not commonly found in single-byte character sets (SBCS).

Before going any further, it might be wise to explain some key concepts for a better understanding of the topic. In general, characters are symbols of a writing system used to represent sounds, syllables or notions, reserved for letters, numbers, and punctuation. While the English language, for instance, is made up of 26 characters and some other 70 symbols required for programming, languages such as Japanese, using three phonetic alphabets, require around 3,000 different characters for daily communication. Computers use a binary system to process and represent characters, which are organised in tables known as code pages. Therefore, each character is encoded with a unique numeric value. For instance, the 7-bit sequence "1000001" corresponds to "A" in the standard American Standard Code for Information Interchange (ASCII) character set. This scheme, with 128 characters, provides support for English language, but it is insufficient for most Western languages, and even more so for Japanese, Chinese or Korean.


Regional and Language Options

Regional Options | Languages | Advanced

Text services and input languages

To view or change the languages and methods you can use to enter text, click Details.

Details...

Supplemental language support

Most languages are installed by default. To install additional languages, select the appropriate check box below.

[✓] Install files for complex script and right-to-left languages (including Thai)

[✓] Install files for East Asian languages

OK | Cancel | Apply

Fig. 1: The Languages tab in the Regional Settings Options

In order to support Western European languages, 8-bit (one byte) character sets were developed to provide 256 character positions. With this additional bit, the ISO (International Standards Organisation) Latin-1 character set implements the ASCII table by adding all characters specific to Western European languages such as accented letters. Similarly, all characters for Central European languages are implemented in the ISO Latin-2 character set. These implementations to ASCII are code-named depending on the language they provide support for. Latin-1 is known as ISO 8859-1 while Latin-2 corresponds to ISO 8859-2. Cyrillic (ISO 8859-5), Arabic (ISO 8859-6), Greek (ISO 8859-7) or Hebrew (ISO 8859-8) are other scripts supported with an 8-bit encoding.

Depending on the input language, computers need to be set up and use the corresponding code page, which interprets keystrokes and displays characters accordingly.

For alphabets with thousands of characters, it is necessary to use a longer binary string that ensures each character is uniquely identified. More precisely, a two-byte string was the most suitable solution for such encoding. Thus, Unicode was developed as a double-byte (16 bits) character set comprising 65,536 different positions, enough to cater for most of the world's languages. Compatibility with smaller code pages is provided, since the first 256 characters are identical to those on ISO 8859-1. Characters from non-Latin scripts above position 256 are grouped by scripts, i.e. Syriac, Bengali or Hiragana.

In computing, as well as in hand-writing, any given character can be depicted in many different ways. The diverse shapes of the same character are known as glyphs, while a set of glyphs comprising all characters for the same alphabet constitutes a font. As a consequence, codepages and fonts are closely related. Each glyph in a font set is assigned a value which corresponds to the character the user types in the keyboard. If the keyboard layout, the code page, and the font do not match, the result can be totally unexpected. Since the same value can correspond to different characters depending on the code page, data transfer cannot be relied upon between computers running different code pages.

## Handling double-byte languages

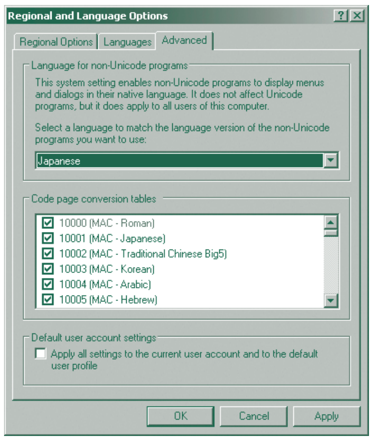
The first thing one might notice when receiving documents in a language like Chinese is that characters on the file name display as squares. Question marks are also a common result if you try to open a Catalyst ttk file that has been translated into Korean. In a Windows environment, it is easy to gain support for Asian languages. Simply go to Control Panel, Regional and Language Options, click on the Languages tab and check the box Install files for East Asian languages (See Fig. 1). Nevertheless, applications like MS Word can emulate glyphs from Asian scripts, even if they are not installed on your system. However, DTP and design software cannot emulate fonts that are not installed on your machine. For further processing of the file, such as printing to a PDF document, proper fonts for the language to be displayed need to be used.

Using computer aided translation software with DBCS languages can also become a bit tricky. Those packages relying on Word as a translation interface (TRADOS, WordFast, MetaTexis) depend on the word processor to support characters and, thus, feed the translation memory. This information is then saved to a file and encoded as Unicode text. Unicode fonts can also be set for the translated segment, so the data will not become corrupted when the document is opened in a machine with a different code page. For instance, with TRADOS, tagged documents (HTML, XML or files exported from DTP applications) are opened in a specific translation interface that converts characters other than those on the ASCII code page to character tags that are later interpreted and replaced by the actual character during the process of generating the final translated file.

## Different output, different approaches

Often, translated RTF files have to be imported back into the native DTP application where the text was generated. This is because most translators cannot work directly on such applications and, therefore, the semi-automated export and import process turns out to be the most efficient procedure. This process, normally accomplished using a specific plug-in for the DTP application, is usually rather straight forward and seldom requires more user interaction than clicking a couple of buttons. QuarkXPress and PageMaker files are two common examples for this kind of localisation engineering work.

Before importing files that have been translated to DBCS languages, a few actions need to be performed. In my experience, the best approach is to set up the language for non-Unicode applications according to the language of the text to be imported. This is done in the Advanced tab, again in the Regional and Language Settings on the Control Panel (See Fig. 2). Choose the corresponding language from the drop-down menu and also set the keyboard layout to the language of the translated file. On doing this, you will ensure all character information is correctly stored on the clipboard during the automated process.


Regional and Language Options

Regional Options | Languages | Advanced

Language for non-Unicode programs

This system setting enables non-Unicode programs to display menus and dialogs in their native language. It does not affect Unicode programs, but it does apply to all users of this computer.

Select a language to match the language version of the non-Unicode programs you want to use:

Japanese

Code page conversion tables

- ☑ 10000 (MAC - Roman)
- ☑ 10001 (MAC - Japanese)
- ☑ 10002 (MAC - Traditional Chinese Big5)
- ☑ 10003 (MAC - Korean)
- ☑ 10004 (MAC - Arabic)
- ☑ 10005 (MAC - Hebrew)

Default user account settings

☐ Apply all settings to the current user account and to the default user profile

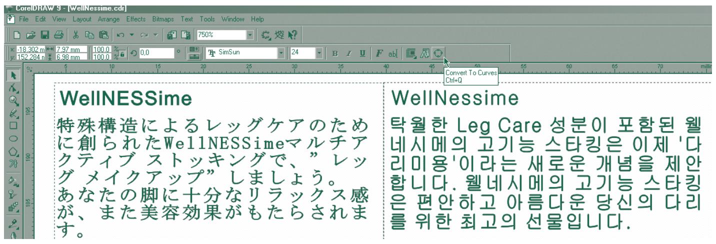
OK | Cancel | Apply

Fig. 2: The Advanced tab, in the Regional and Language Settings on the Control Panel

On the other hand, design applications –such as Freehand or Corel Draw– are not so "export/import friendly" and lack such plug-ins. This situation makes it necessary to manually copy and paste text from one application into another. In such a case, the above-mentioned settings should also be used. Creating a Unicode plain text file containing the translation would ensure all characters are correctly transferred during the copy-and-paste process. It is also a good idea to set the font in that text file to one that is also available in the design software, so it can be used after pasting. Although apparently the copy and cut functions work exactly the same regarding clipboard content, for certain applications, the text is correctly pasted only when the cut function is used. After changing the pasted text to the same font as the one used in the text file, all characters should be displayed properly. You can even place columns of text with different scripts in the same document, a common requirement for product brochures, boxes or pamphlets that will be distributed in multiple Asian languages.

Design applications feature the "convert text to curves" option, a quite convenient process that turns glyphs into curves –a layer that is similar to an image (See Fig. 3). This conversion eliminates all character information, so the text can no longer be modified. However, the resulting layer containing the text can be resized, duplicated or moved without any loss of quality. Also, this format is platform independent, so no conversion issues should arise if the printer is using a Mac machine, which is not unusual. From a Freehand or Corel Draw file it is possible to produce a separated image file such as EPS or TIFF for example, which can be used after as an independent object that may also be embedded in any other document. This is particularly interesting when the client has no support for Asian languages and just wants to send the document for printing without any further ado.

Sometimes, the client deliverable includes final QuarkXPress files together with a PDF document usually for reference, but often for Web viewing or e-distribution. The printing process of files with DBCS languages has two basic requirements: the computer must have Asian language support installed and the fonts used in the document must be available. It is important to use True Type fonts, since these would allow us to print the exact layout that can be seen on screen. Adobe Acrobat, the most popular PDF publishing application, features two printing methods: PDF writer –for quick document printing– and Distiller –for accurate printing jobs. The latter allows the user to specify more conversion settings and store them in profiles for a better management of client specifications (See Fig. 4). Font embedding is the key feature for a successful PDF printing job in non-Latin languages. Therefore, all fonts used in the document must be embedded. Also pay attention to correctly updating all images in the document and using a down sampling rate that would render colour images to a good quality in the final PDF.


CorelDRAW 9 - [WellNessime.cdr]
File Edit View Layout Arrange Effects Bitmaps Text Tools Window Help
750%
SimSun
24
Convert To Curves Ctrl+Q

WellNESSime

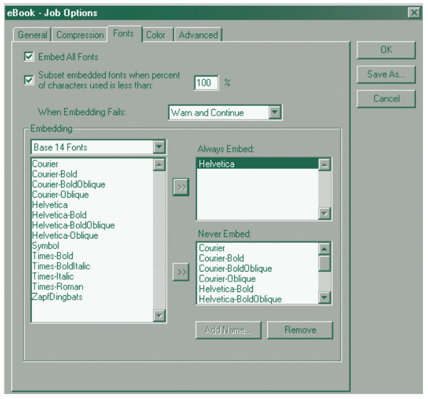
特殊構造によるレッグケアのために創られたWellNESSimeマルチアクティブ ストッキングで、" レッグ メイクアップ" しましょう。あなたの脚に十分なリラックス感が、また美容効果がもたらされます。

WellNessime

탁월한 Leg Care 성분이 포함된 웰네시메의 고기능 스타킹은 이제 '다리미용'이라는 새로운 개념을 제안합니다. 웰네시메의 고기능 스타킹은 편안하고 아름다운 당신의 다리를 위한 최고의 선물입니다.

Fig. 3: The "Convert To Curves" option in Corel Draw

When it comes to resource files, internationalisation requirements need to be met prior to starting the translation stage. Realising that DBCS language strings are not supported by the core components of the application once translation is finished would not be of much use. As a general rule, it is advisable to run smoke and cosmetic tests at the project evaluation stage in order to be able to determine these particulars. When using freelance translators for translating software files into DBCS languages, it is wise to ask for Unicode files in the project instructions. Quite often, translators are not aware of internationalisation issues and believe that if the file displays correctly on their machine, it will in any other computer, which in most cases proves to be false.


eBook - Job Options

General | Compression | Fonts | Color | Advanced

☑ Embed All Fonts

☑ Subset embedded fonts when percent of characters used is less than: 100 %

When Embedding Fails: Warn and Continue

Embedding

Base 14 Fonts

- Courier
- Courier-Bold
- Courier-BoldOblique
- Courier-Oblique
- Helvetica
- Helvetica-Bold
- Helvetica-BoldOblique
- Helvetica-Oblique
- Symbol
- Times-Bold
- Times-BoldItalic
- Times-Italic
- Times-Roman
- ZapfDingbats

Always Embed:

- Helvetica

Never Embed:

- Courier
- Courier-Bold
- Courier-BoldOblique
- Courier-Oblique
- Helvetica-Bold
- Helvetica-BoldOblique

Add Name... | Remove

OK | Save As... | Cancel

Fig. 4: Font embedding.

Localisation of documentation and software into DBCS languages requires specific engineering work, settings and procedures. It is necessary to schedule enough time for engineers to analyse the process prior to starting the project –preparation– as well as after the translation stage –implementation. Ideally, once the workflow for a specific client has been fully defined and tested, preparation and implementation times should decrease although some contingency time may be allowed in case unexpected problems arise.

*Aitor Medrano graduated in Translation and Interpreting at the University of Granada, Spain, and received a Diploma with Honours in Software Localisation at the University of Limerick. Specialising in Web i18n, multimedia contents, software engineering and CAT tools, he is a localisation consultant and a certified TRADOS expert at CPSL. He also works as a freelance translator of sci-tech documentation, software and multimedia products. He can be reached at amedrano@celerpawlowsky.com*