Displaying Chinese Characters In Blaise

Gina-Qian Cheung & Youhong Liu
Institute for Social Research, University of Michigan

1. Introduction

A Blaise data model can be multilingual. This is important for interviewing multi-language populations. One of the projects conducted by the Institute for Social Research at the University of Michigan is the National Latino and Asian American Study (NLAAS). There are five eligible languages in the study – English, Spanish, Chinese, Vietnamese, and Tagalong (Philippine native language).

While three of the languages are based on Latin characters, Vietnamese and Chinese are much different and more difficult to handle. For Vietnamese, it is required to install a Vietnamese font on the computer and then set a font representation in Blaise mode library that point to this Vietnamese font. For example, you can set @V =Vptimes. In Blaise field text, we can just put @V around the Vietnamese text. There are many details about how to prepare a Vietnamese document to be used on Blaise field text and how to do Vietnamese text fills etc. This will not be discussed in this paper.

In this paper, we will concentrate on Chinese language, one of the five languages utilized in the study. We will present how to use the resources available to add Chinese capability to the Blaise Environment. Please note that we only describe one of the possible approaches on the Windows platform (English version), other alternatives also exist.

2. Chinese Encoding Methods

First, we need to give some background about the basis of Chinese encoding system.

Encoding involves mapping a character to a numeric value so that the character can be identified through its associated numeric value. Computer systems process data in terms of bits, the most basic units of information processing. Bits are mapped to the 1 (on) and 0 (off) and are grouped together into units called bytes. Bytes can be composed of 7 or 8 bits. 7-bit bytes can allow up to 128 unique combinations while 8-bit bytes up to 256 unique combinations. While these numbers are good enough for encoding most writing systems of Western languages (such as the ASCII character set), with tens of thousands of distinct characters, they are far from enough to represent the writing system of Chinese (and those of other East Asian languages such as Japanese and Korean). The solution to this problem is to use multiple bytes to represent a single character. For example, an 8-bit 2-byte system can encode up to 65,536 (256x256) characters. This is known as a Double-byte Character Set (DBCS).

Two major computer encoding methods are used for Chinese: Big5 and GuoBiao (GB). Big5 encodes traditional characters and is used in Hong Kong and Taiwan, while GB encodes simplified characters and is used in Mainland China and Singapore. Another encoding method, Unicode, can be used in most of the world's major languages, including both simplified and traditional Chinese.
The computer does not know how to display the Double Byte characters. If you open a document that was created using one of these encoding methods on your

English Windows computer, all you will see is a bunch of gibberish. To display the characters properly, you will need a localized version of Windows (e.g. Chinese Windows System) or a software program (such as TwinBridge or NJStar), which acts like an interpreter – watching for double-byte characters and jumping in and converting the encoded characters into Hanzi (Chinese characters) as they come up.

3. Our Approach to Displaying Chinese in Blaise

There are several approaches to handle Chinese on computers. One technique is to have the entire operating system support Chinese. This is the most popular option in which the user only deals with Chinese and no other languages. Microsoft sells both traditional and simplified Chinese versions of its Windows operating system.
In this case, it was not feasible to use a Chinese operation system, because our users are primarily English based. Our approach is to use a program to add Chinese capability to Blaise. We chose NJ Star, but there are many other programs available in the market, such as Twin Bridge and Chinese Star, etc.

4. Steps to Input Chinese Word Specs to a Blaise Source File

Suppose we have an existing Blaise English source file, if we have a corresponding Chinese word document, how do we input Chinese into the source file? We cannot copy and paste the Chinese text directly from the word file to the Blaise editor. If we do a direct copy and paste, all the Chinese characters will become "?????????" in Blaise file editor because of the encoding problem. In order to input a Word file into a Blaise source file, we need to do the followings:

1. Converting the MS-Word document to an encoded text file

    a. On the File menu, click Save As
    b. In the File name box, type a new name for the file
    c. In the Save As type box, select Encoded Text
    d. Click Save
    e. Click Yes to discard any formatting and save as text
    f. Select Other Encoding, and then select "Chinese Simplify (GB2132)" or "Chinese Traditional (Big 5)" in the list. Please note that you should choose one of the two encodes according to the file encoding used, i.e. the input encoding used for the file. If you select "Chinese Simplify (GB2132)", but the file uses Big 5 encoding, the result will be wrong. You can also preview the text in the Preview area to check whether it makes sense in the encoding standard you have selected (If the Preview area is not visible, click Show Preview).

2. Adding space between Chinese characters

    The Blaise DEP expects to find spaces to break the text into words and lines. Since the encoded Chinese file converted from MS-Word does not use spaces, the Blaise DEP may break up the Chinese text in inappropriate places, including in the middle of a double-byte character, or not break the line at all, leaving a large run of text off beyond the edge of the window. Actually, the early versions of Netscape and Internet Explorer had this kind of problem. To help proper formatting of Chinese text, a space is needed between each character; this will allow Blaise Dep to find an appropriate place to break the text into lines. Most Chinese software programs have a function to add a space in between the characters. We used NJ Start Communicator to conduct this task.

3. Inputting the Chinese text into the Blaise source file

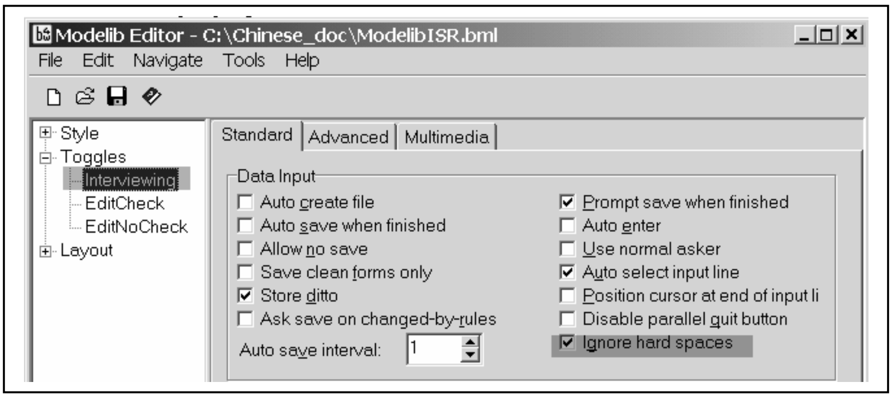
   Since it is time consuming to copy field by field to the Blaise source file, UM developed a utility - Foreign Language Merge Utility (FLMU) - that helps to insert Chinese into the Blaise source code. This utility can also be used for any other languages. Using this utility can save a substantial amount of time.

4. Handling Blaise Control Characters

   In the Blaise Dep, character "@" is a font or format indicator and "^" precedes the field or the variable name whose value will be included in the field text. In the Chinese encoding, "@" and "^" both can be part of a Chinese character. In this case, we must type them twice. We can do "Find and Replace All" to the text file before we use the FLMU to insert text into the Blaise source code. It is still difficult to remember typing these two characters twice when manually inserting Chinese text. Later, it was found that "@" and "^" are used in Big 5 encoding, but not in GB encoding. So we decided to use GB encoding for our instrument. This made our job far easier. If the documents we received use Big 5 encoding, we can use NJ Start Communicator to convert them from Big 5 to GB encoding.

   To use a hard space in Blaise field text, normally we can type <Ctrl-period> in the Blaise editor. Unfortunately, this character is also used by some of Chinese characters, both in Big 5 and GB encoding system. If we type <Ctrl-period> twice, it will only instruct Blaise Dep to display two spaces. To solve this problem, in Blaise Mode Library, we can choose to ignore hard spaces (Figure-1). So this means, in a Blaise instrument that includes Chinese characters or any character set that has <Ctrl-period> character, we can no longer use hard spaces in field text.

Figure 2 – Ignoring hard spaces in Blaise Mode Lib



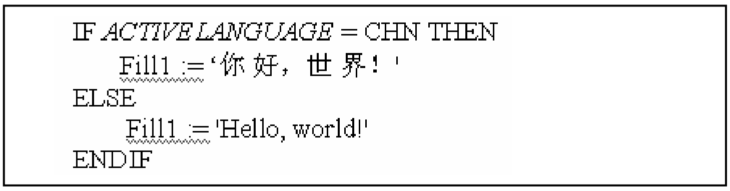
Modelib Editor - C:\Chinese_doc\ModelibISR.bml

Data Input

□ Auto create file
□ Auto save when finished
□ Allow no save
□ Save clean forms only
☑ Store ditto
□ Ask save on changed-by-rules
Auto save interval: 1
☑ Prompt save when finished
□ Auto enter
□ Use normal asker
☑ Auto select input line
□ Position cursor at end of input li
□ Disable parallel quit button
☑ Ignore hard spaces

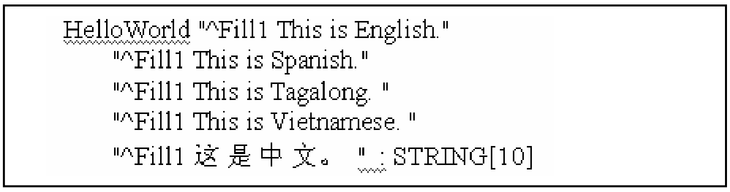
5. Chinese Fills in Blaise

In Blaise, fill holds the value of another field, and inserts it into the question text. You can test the current language with the key word *ACTIVELANGUAGE.* You might use the following example to determine which fill to use in language text:



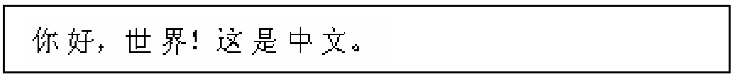
```
IF ACTIVE LANGUAGE = CHN THEN
    Fill1 :=‘你好，世界！'
ELSE
    Fill1 := 'Hello, world!'
ENDIF
```

Where Fill1 may be a fill in a question text:



```
HelloWorld "^Fill1 This is English."
    "^Fill1 This is Spanish."
    "^Fill1 This is Tagalong. "
    "^Fill1 This is Vietnamese. "
    "^Fill1 这是中文。 " : STRING[10]
```

The Chinese question text display for the field HelloWorld will be:



你好，世界！这是中文。

6. Switch Language During Interview

Blaise provides a default menu entry for language switching. When a case is suspended during the interview, it will use the first language the next time the case is invoked again not the language you used last time. This is not desirable. We decided to use a parallel block to switch languages. In this parallel block, SETLANGUAGE command is used to switch between languages. This works very well, not only we can keep the suspended language setting, but also store the language being used during interview in a array so that the data can be used for analysis later.

It is possible that the parallel language-switching block may create a conflict with the default menu language switching. To avoid this conflict, the default language switching can be disabled (Figure 2). A menu item can be added that invokes the language switching parallel block (Figure 3).

Figure 3 – Disabling Default Language Switch in Blaise Menu

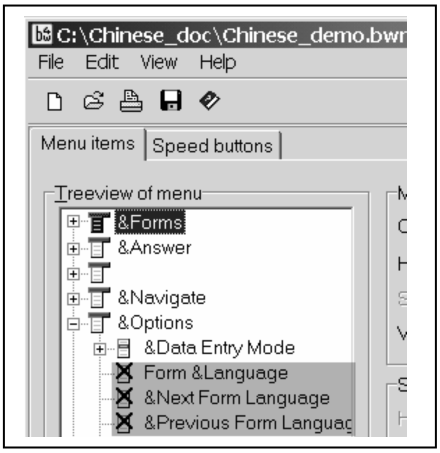
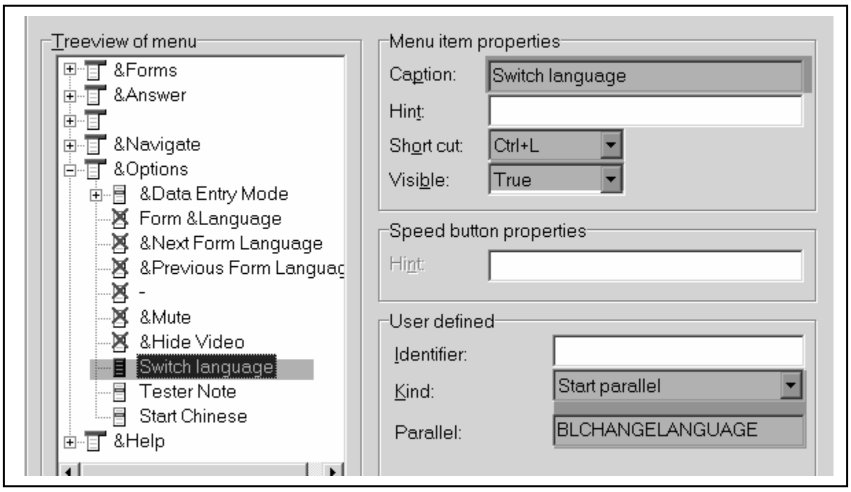


Figure 3 – Adding Parallel Language Switch Block in Blaise Menu

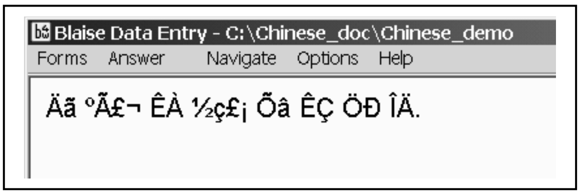


Caption: Switch language
Hint:
Short cut: Ctrl+L
Visible: True
Hint:
Identifier:
Kind: Start parallel
Parallel: BLCHANGELANGUAGE

7. Setting Menu Item For Chinese View

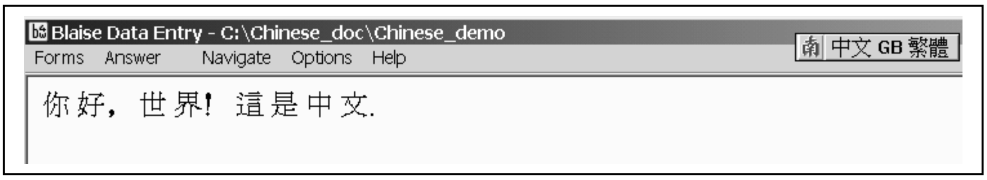
During a Chinese interview, the interviewer needs to switch the interview language to Chinese, but also has to turn the Chinese Viewer on. Figures 4 and 5 show the difference with and without a Chinese Viewer. To allow quick access to the Chinese Viewer, a menu item and a hot key can be added, e.g. F6 to enable the Chinese viewer (Figure 6).

Figure 4 – Displaying Chinese without Chinese Viewer


Blaise Data Entry - C:\Chinese_doc\Chinese_demo
Forms Answer Navigate Options Help

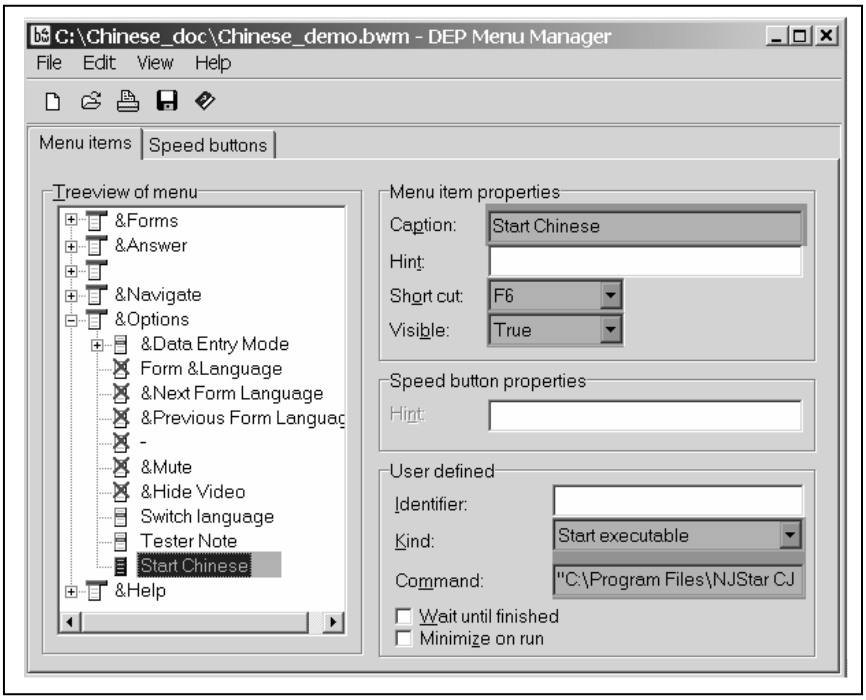
Ää °Ã£¬ ÊÀ ½ç£¡ Õâ ÊÇ ÖÐ ÎÄ.

Figure 5 – Displaying Chinese with Chinese Viewer


Blaise Data Entry - C:\Chinese_doc\Chinese_demo
Forms Answer Navigate Options Help
南 中文 GB 繁體

你好，世界！這是中文.

Figure 6 – Adding Chinese Viewer Access to Blaise Menu



Caption: Start Chinese
Hint:
Short cut: F6
Visible: True

Hint:

Identifier:
Kind: Start executable
Command: "C:\Program Files\NJStar CJ
☐ Wait until finished
☐ Minimize on run

8. Summary

Displaying Chinese in Blaise is different from other languages because it requires a two-byte encoding system, but it is not a difficult task if the proper tools are used and the right procedures are followed. At the University of Michigan, the multi-language study, National Latino and Asian American Study (NLASS) was deployed in May 2002. The interface was well received by both field staff and the study managers.

9. References

Statistics Netherlands Blaise Developer’s Guide
http://users.erols.com/eepeter/chinesecomputing/encodings/
http://www.njstar.com