Proceedings of the 11th Seminar of the International Association for Tibetan Studies, Königswinter, Germany, Aug 27-Sept 2, 2006 — Information Technology Panel

Tibetan Information Technology: New Era and New Challenges

Tashi Tsering
China Tibetology Research Center
Beijing, China
ttsering1@gmail.com

Abstract

While most people in the world have been enjoying computer technology and the Internet for decades, the Tibetan community still does not have an operating system that supports the Tibetan language. At the present, there are many word processors that do not communicate with each other, and there is no single computer system yet in the world that supports the international standard, Unicode for Tibetan. After nearly ten years of efforts by the Tibetan computing community, a suitable computing foundation for Tibetan language is only now becoming a reality as we enter a new era of Tibetan information technology. This paper, after a brief introduction to the current Tibetan computing situation, will address the new font technology, OpenType, which makes the implementation of Tibetan Unicode possible in major popular operating systems such as Microsoft Windows, Macintosh OS, and Linux. Since Microsoft Windows is overwhelmingly the most popular computing environment among the Tibetan community in China, this paper will focus on introducing the new Microsoft Windows operating system, Windows Vista, which will support the Tibetan language and will be released at the end of 2006. The second part of this paper will focus on the challenges that will still need to be addressed even after the advent of Tibetan Unicode supported operating systems. Specifically, these are: (1) how to develop a better Tibetan OpenType font; (2) how to design a more efficient Tibetan keyboard based on Tibetan Unicode; (3) understanding what will be the major technology issues in developing software for Tibetan language. In addition to these topics, this paper will address another major challenge that faces the community — standardizing computer terminology in Tibetan. This is another fundamental issue of Tibetan computing, and we report on a related project that has already begun to address this.

Introduction: The Current Tibetan Computing Situation

While most people in the world have been enjoying computer technology and the Internet for decades, the Tibetan community still does not have an operating system that supports the Tibetan language. At the present, there are many word processors that do not communicate to each other, and there is no single computer system yet in the world that supports the international standard, Unicode for Tibetan. At present there are many

"Tibetan systems" or "Tibetan word processors" and they do allow us to process Tibetan in the computer. These systems are daily used to process Tibetan documents, publish Tibetan books, and even build Tibetan web sites. The real situation however, is that there are:

- too many encodings for Tibetan: Beida Founder Tibetan encoding, Huaguang Tibetan encoding, Tongyuan Tibetan encoding, Bandrida Tibetan encoding, and many others around the world;
- too many Tibetan fonts using or "borrowing" code cells of Latin characters with different encodings or different mappings from Tibetan characters or stacks to those code cells: fonts such as Sambhota, Tibetan Machine, TCRC, Jamyang, etc.

Because of these incompatible encodings and fonts developed independently and over many years, the result has been a mess of Tibetan word processors that do not communicate with each other in terms of Tibetan encodings for data exchange. Since there is no single formal computer system yet in the world that supports the international standard of Unicode for Tibetan, it has still not been possible to establish a suitable computing foundation for the Tibetan language based on Tibetan Unicode.

Although the Tibetan language was assigned Unicode code points almost ten years ago, Tibetan Unicode is still not being used. The principle reason for this was the challenge posed by the complexity of the Tibetan language in terms of the word processing capacity of a computer. Now that smart font technologies such as OpenType have been introduced, implementation of Tibetan Unicode at the operating system level of a computer has become possible.

OpenType Font Technology

The challenges of typography

The invention of typography has been called the beginning of the Industrial Revolution. However, there were still many challenges that people faced when working on more complex scripts. These challenges of typography were:

- limited sets of glyphs coupled with predefined shaping logic
- language-specific predefined shaping logic

In the past, even though sometimes one script was shared by more than one language, many languages were not handled at all. Apple GX's birth brought about a new era: the era of smart font technology. These "smart fonts" included fonts with such technologies as:

- Apple's AAT (Apple Advanced Typography)
- Summer Institute of Linguistics' Graphite technology
- OpenType – Microsoft and Adobe

Smart font technology can be defined as a font format that allows the type designer to specify additional information about the way glyphs are used in the font. This allows the traditional one character equals one glyph model in a font to be rendered obsolete. It also makes it possible for multiple glyphs to represent a single character (known as contextual or alternate forms) or a single glyph to represent a group of characters (known as ligatures).

There are two approaches to providing typographic support through smart fonts. AAT and Graphite require the font developer to put specific language handling logic in each of the

fonts. In contrast, OpenType requires the common language handling to be dealt with outside of the font while keeping typographic control inside of the font. Microsoft, for example, places the language logic in the Uniscribe engine. This approach allowed OpenType font developers to focus more on creating good typefaces and less on the language handling logic required for the font to work.

What is OpenType Font Technology?

Basically, OpenType Font Technology is characterized by:

- an architecture for supporting complex scripts and advanced typography, and
- fonts that are Unicode-based, allowing a rich mapping between characters and glyphs

It is a font technology developed jointly by Adobe and Microsoft as an extension of the TrueType font format that also can contain PostScript data. OpenType fonts are cross-platform, that is, the same font file works under both Macintosh and Windows operating systems. This digital type-format offers extended character sets and more advanced typographic controls. Like TrueType, it is a single file containing all the outline, metric, and bitmap data for an OpenType font. Adobe has converted all of their typefaces into OpenType format.

The Advanced Typographic tables (also called OpenType Layout tables) extend the functionality of fonts with either TrueType or PostScript outlines. OpenType Layout fonts contain additional information that extends the capabilities of the fonts to support high-quality international typography. It allows font creators to design better international and high-end typographic fonts.

The OpenType Layout tables contain information on glyph substitution, glyph positioning, justification, and baseline positioning, enabling text-processing applications to improve text layout. For complex languages such as Tibetan, with its OpenType Layout tables, OpenType Font Technology makes it possible for them to be implemented under Unicode.

In conclusion, OpenType Layout fonts:

- allow a rich mapping between characters and glyphs, supporting ligatures, positional forms, alternates, and other substitutions,
- include information to support features for two-dimensional positioning and glyph attachment,
- contain explicit script and language information, so that a text-processing application can adjust its behavior accordingly, and
- have an open format that allows font developers to define their own typographic features.

Tibetan Unicode: One Tibetan in All Computers

Unicode is the foundation of global computing. It is a basic standard of the information technology industry. Without this standard or by not following this standard, different parties will have problems exchanging their data.

As previously stated, there has been a confusion of different Tibetan encodings for decades, with the resulting problem of exchanging data within the Tibetan community. At the present time, there is no computer software in the world that supports the Tibetan language based on Unicode. This is simply because of the technical difficulty of implementation of Tibetan Uni-

code in computer operating systems. Fortunately, after it was introduced, the OpenType font technology made the implementation of Tibetan Unicode possible in major popular operating systems such as Microsoft Windows, Apple's Mac OS, and Linux. Microsoft has already prepared its new Windows OS, Vista, which fully supports Tibetan Unicode along with many other complex languages; Vista will be released by the end of 2006. Now it is time for the Tibetan community to embrace this new era of Tibetan computing. The dream of one Tibetan data encoding in all computers is coming true soon. We will soon be able to process Tibetan in computers and on the Internet as easily as English and the many other languages in the world.

The only challenge to be faced after the release of Tibetan-supporting Unicode systems will be the migration of legacy-encoded Tibetan data to Unicode. Tools will be needed to convert that legacy data into the Unicode encoding. Fortunately, we have already prepared for such event. Many tools for converting data from legacy encodings to Unicode have been developed. One of them is the Universal Tibetan Font Converter, which was developed by the author under the support of the Trace Foundation in 2004. It provides users with the functionality to convert data from eleven legacy encodings to Unicode. It is free software.

How Windows Vista Supports Tibetan

Microsoft Windows is overwhelmingly popular among Tibetan people in China. Also, Microsoft is the only company presently slated to provide Tibetan Unicode supported in its OS product, Windows Vista. Consequently, I would like to share here the basic idea of how Windows Vista supports Tibetan Unicode.

Windows Vista supports Tibetan at operating system level or at locale level based on Tibetan Unicode. The Tibetan system in Windows Vista consists of the following five parts:

- Tibetan OpenType Font: Microsoft Himalaya
- Tibetan Keyboard
- Tibetan Uniscribe Engine
- Tibetan Locale Information
- Tibetan Sorting Algorithm

Microsoft's default Tibetan font in Windows Vista is called Microsoft Himalaya. It is an OpenType font. Unlike legacy Tibetan fonts, this font can handle arbitrary depth Tibetan stacks. What this means is that a user can type any arbitrary Tibetan stack, syllable, or word; it is an open-ended font system. With the MS Tibetan keyboard and MS Tibetan OpenType font Microsoft Himalaya, users can easily process Tibetan language based on Tibetan Unicode in Windows Vista.

"Tibetan Uniscribe" is a portion of the Windows Uniscribe engine that processes Tibetan characters and together with the OpenType tables and glyphs in the font, renders Tibetan text in precisely the way users expect it to. The Uniscribe Tibetan shaping engine works in following steps:

- Put Unicode Tibetan input into clusters of characters that should be kept together
- Map the clusters of Tibetan characters into the nominal glyphs in the font
- Apply the lookups in the font that substitute or position the glyphs

Tibetan Uniscribe also contains the control information for Tibetan line justification, alignment, and many other culturally related features.

Tibetan locale information is responsible for coordinating all Tibetan language-related behaviors inside Windows. The implementation of a Tibetan sorting algorithm brings the added feature of sorting Tibetan words, enabling work on a Tibetan dictionary, etc., completely freeing end-users from having to sort Tibetan words manually; in Microsoft Office 2007, it is anticipated that users will be able to sort a Tibetan dictionary in seconds.

Developing Tibetan OpenType Fonts

After Windows Vista is released, it will be possible to use more and more Tibetan fonts based on Tibetan Unicode. As a result, we believe that more and more Tibetan OpenType fonts will be designed and published. With OpenType font technology, it will be possible to design not only Tibetan Dbu-Can fonts, but also other typefaces of Tibetan fonts, such as Tshugs-Ma, 'bru-Tsha, and 'khyug-Yig. It is the powerful feature of OpenType layout, particularly the Substitution and Positioning features, that allows for the design of very efficient, small-sized and simple, but very powerful Tibetan fonts such as the Microsoft Himalaya Tibetan font.

Here we would like to share the basic requirements, the design idea, and design strategy of the Microsoft Tibetan OpenType font Microsoft Himalaya. A discussion of the glyph and ligature design would be more about artistic style and not a technical issue (and thus beyond the topic of this paper). Rather, by providing here the same technical solution, end-users can design their own very efficient Tibetan font by following the design idea and design strategy of Microsoft Himalaya.

The principle behind designing Microsoft Himalaya was to treat those important and non-important, frequently used and infrequently used Tibetan characters and stacks in different ways by giving those important and frequently used characters and stacks high priority, while using a dynamic stacking procedure to process those less important Sanskrit transliteration stacks that have very low frequency of usage.

The basic requirement or specification set for Microsoft Himalaya was that:

> The font file should be able to display any arbitrary sequence of Tibetan Unicode code points correctly. In other words, the font file should allow user to type any Tibetan character, stack or word, including any Sanskrit transliteration stacks, and even any stack and sequence of characters that are neither Tibetan nor Sanskrit transliteration.

To meet this specification, this design idea and design strategy was followed:

1. First the entire set of Tibetan characters and stacks was separated into two sets based on their frequency data. All of the 193 Unicode Tibetan characters and all of the 172 Tibetan stacks without top vowel signs were put into Set A, plus 98 stacks consist of 30 constants combining with 'A-chung, Ya-rtags, Ra-rtags and Wa-zur respectively, plus some amount of the more frequently used Sanskrit transliteration stacks. The rest of the stacks were Sanskrit transliteration stacks. Since the number of these stacks is indeterminant, they were put into Set B, which should be an open set.

   Those more frequently used Sanskrit transliteration stacks to be put into Set A were chosen based on setting a cumulative cut-off frequency for all characters and stacks including Tibetan stacks at 99.99%. This meant that theoretically, in a Tibetan file with

ten thousand non-combining characters and stacks, there would be no more than one less-frequently-used Sanskrit transliteration stack contained in Set B. As a result, there are 102 Sanskrit transliteration stacks that fall into set A.

2. In order to have better typeface results, precomposed glyphs for stacks falling into Set A were created. These precomposed glyphs are then displayed using the Substitution feature of the OpenType layout. For those stacks falling into Set B, dynamic stacking is used by applying the Positioning feature of the OpenType layout. In order to do dynamic stacking, 622 varieties of glyphs as parts for assembling at different levels of stacks were created.

In total, the font file of Microsoft Himalaya has only 1,236 glyphs contained in it. Nonetheless, the performance of this font is nearly perfect. It can display arbitrary stacks or strings of Tibetan characters. There is no Tibetan word or Sanskrit transliteration stack that a user can not type. Theoretically, the depth of stack level is unlimited. Users can type arbitrarily high stacks, just as users can type arbitrarily long strings of Latin letters. At present, no Tibetan font in the world other than Microsoft Himalaya displays such performance.

Designing a Tibetan Keyboard Layout Based on Tibetan Unicode

Keyboard layout is the crucial issue of a keyboard design for a language in the computer. Some have argued that a keyboard layout should be designed based on the frequency data of the characters of the language. Such a keyboard design based on frequency data is believed to make users very comfortable with the keyboard, as well as making the keyboard easy to use.

Just as with legacy Tibetan fonts and their encodings, there are many keyboard layouts for Tibetan language at use in the world as well. Two of these are from Chinese companies, and people in China are somewhat familiar with them. Within the Tibetan academic community, people are more familiar with the keyboard layout based on the Wylie transliteration system called, accordingly, the Wylie keyboard.

In moving to Tibetan Unicode, it is necessary to switch these keyboards from legacy encodings to Unicode as well. However, implementing these keyboard layouts for the Unicode Tibetan character set can be very difficult. In order to be able to input all of the Unicode Tibetan characters, a new design for a Tibetan keyboard layout that covers all 193 Unicode Tibetan characters is needed.

In 2005, Tibetan computer experts in China gathered together a team in order to design a Chinese national standard Tibetan keyboard layout under the support of the Chinese government — a keyboard layout that has already been implemented in Microsoft Windows Vista. The keyboard layout completely followed the universal keyboard layout design principle, which was to assign one key for one character.

To assign 193 characters to 193 keys, five virtual keyboards were needed. The 193 Tibetan Unicode characters were thus spread out over five virtual keyboards based on the following principles:

1. following the principles of the Dvorak keyboard layout by considering the frequency data of Tibetan Unicode characters on the same keyboard;
2. considering the Wylie system when assigning characters on the same keyboard;
3. keeping a letter, its combining form, and its other varieties constant on the same key locations of different keyboards, to make them easy to remember and easy to type;

4. putting the most frequently used characters on the first keyboard, less frequently used ones on the second keyboard, then the third keyboard, and so on;
5. designing in a manner such that no details of implementation of the keyboard layouts would be predefined in the standard, leaving users the full right to determine how to implement their keyboards.

These five layouts were assigned to five virtual keyboards: the default keyboard, keyboard with dead key "m", keyboard with SHIFT, keyboard with ALT+CTRL+SHIFT, and keyboard with dead key "M". A dead key followed by a regular key will generate a new key code.

It is our belief that by designing these keyboards based on two important factors (frequency data and the Wylie transliteration system) that they will be optimum keyboards for Tibetan Unicode characters, since this design will both guarantee users' ease of recollection of the keyboard layout, and guarantee that users can input all Tibetan words and Sanskrit stacks. For those users who wish to have their own keyboard layouts however, Microsoft has provided tools for such development. The Microsoft tool, MSKLC ("Microsoft Keyboard Layout Creator") allows users to design their own keyboards easily without writing code in a WYSIWYG environment. Such user-designed keyboards can then be installed in Windows Vista to replace the system keyboard.

Developing Software for Tibetan Language

In the past, there was little opportunity to develop software for processing the Tibetan language. Because older systems were not using the Unicode standard code points for Tibetan, but rather were using encodings that "borrowed" code points actually belonging to other languages, it was not possible to develop universal applications for Tibetan prior to support for the Tibetan Unicode character set in operating systems. So long as we have an operating system that supports Tibetan Unicode, we are free to develop software for the Tibetan language based on the Unicode standard.

Developing software for Unicode Tibetan is no different than for any other language. For every programming language, the same rules of programming for Unicode as for other natural languages are applied to Tibetan. For example, for C/C++ programming language, the same header files that have Unicode features at the beginning of the code are included, such as

```
#include <wchar.h>
```

and declare Unicode in the code with

```
#define _UNICODE
```

Similarly, for all functions in the code, one must call their "wide character" version, being the Unicode versions of the functions, rather than their legacy versions. For example:

use function ***wfopen(buffer, "rb")*** instead of ***fopen(buffer, "rb")***
use function ***wprintf("No such a file: ")*** instead of ***printf("No such a file: ")***

and always using Unicode data types when defining character data. For example:

use function ***wchar_t UnicodeCharacters[10]*** instead of ***char_t Characters[10]***

The challenge potentially faced in moving to Tibetan Unicode is the inability of those exist-

ing pieces of software to display Unicode Tibetan. It is a problem for any older version of software not compiled based on Unicode, or based on functions that introduce OpenType features. Although a major issue for developing software that processes Tibetan language, we believe it will be just a temporary issue. So long as those older versions of software are updated to meet the specifications of the new operating systems, these problems will disappear.

Standardizing Tibetan Computer Terminology

As commercial OS products such as Microsoft Windows that support Tibetan Unicode at operating system level approach wide-spread distribution, there is another issue standing in the way of fully supporting the Tibetan language in a computer environment: the issue of Tibetan computer terminology. According to Microsoft, there is no technical difficulty behind fully supporting the Tibetan language in Windows at the interface level. The issue is whether there is a standard for computer terminology in Tibetan.

Since such a standard for Tibetan computer terminology does not exist, beginning in 2004 preparations were made for a project to work on a Chinese national standard for Tibetan IT terms. In May of 2005, the research project for a Chinese national standard for Tibetan computer terminology was set up, and funded by the Chinese government under the supervision of the China Tibetology Research Center; it will conclude by the end of 2006. A trilingual dictionary of information technology terms (in Tibetan, Chinese, and English) will be published at the end of that year. The Chinese national standard, "Information Technology—Vocabulary—Tibetan," will be submitted for approval at the same time. It is expected to be approved and published in early 2007.

The major task of the project is to translate international standard Information technology—Vocabulary—(ISO/IEC 2382-1~34) into Tibetan. Its corresponding Chinese national standard for Chinese language is GB/T 5271.1~34. Since the process of updating ISO standards and Chinese GB/T standards is very slow, many parts of the standards are very old, and many new terms are not included in these standards, in particular, many software interface terms. Consequently, it was decided to add terms that were not included in the standards by searching for those terms in Microsoft Windows and Office glossary documents. There are around 4,400 terms in ISO 2382 and GB/T 5271. To this vocabulary set we added 2,800 new terms from Microsoft glossary documents. In total, we have 7,200 information technology terms in our project to be translated and set as a standard.

The approach to the project was to collect four different versions of translations of these terms from project partners in the TAR, Amdo, Kham, and Beijing. A project experts meeting was then convened to discuss and to select the best translation for each term. For terms for which there were no good translations in the four preliminary versions, the project experts group put forth their own translation. After the first project group meeting, a first version of the terminology standard was produced. After several months of review by all project members, a second project experts meeting was convened. At this meeting, each term was reviewed one-by-one once again following which agreement was reached on the translation of 99% of all terms. At that point, 1% of the terms (about 80 terms) remained either without consensus or of questionable translation. In order to seek different opinions about the translations from the community, the temporary result of the project was posted on the project website in order to seek translation suggestions from outside experts.

From the beginning of the project up to now, there have been nearly sixty Tibetan language

and computer experts involved in the project from Beijing and all Tibetan areas in China. About forty experts worked on the first four translation versions, while another twenty experts worked on the project as a group to discuss and then to finalize the translation for each term. Among the twenty experts in the group, seven of them were Tibetan computer experts who were teaching computer science, math, or physics at the college level using the Tibetan language. Four other experts were professional translators. The rest of the group members were Tibetan language experts. It is anticipated that by the end of 2006, the translations of the terms will be finalized and the project concluded.

Conclusion

It is time for the community to bid farewell to non-Unicode Tibetan encodings. The era of Tibetan Unicode computing is coming and in the very near future, processing Tibetan language data in computer systems will reach the same level as many other languages such as English and Chinese. It is our firm belief that in just a few years, when Tibetan children turn on their computers, they will interact with them in the Tibetan language.

Acknowledgements

My thanks go to:

- The organizers of IATS 2006 for their invitation, and members of IATS committee for their approval to set up this panel for us;
- China Tibetology Research Center for funds to attend this conference;
- Microsoft Corp for giving me the chance to design their Tibetan Uniscribe, Tibetan OpenType font and Tibetan keyboard for Windows Vista;
- China Tibetology Research Center again for encouraging me to manage the Tibetan IT Term Project and funding the project;
- Yi Fan from Microsoft China Technology Center for allowing me to use some of his slides in my presentation.

References

1. OpenType Specification (version 1.4) http://www.microsoft.com/typography/otspec/ and http://www.microsoft.com/typography/specs
2. Developing OpenType Fonts for Arabic Script: http://www.microsoft.com/typography/OpenType%20Dev/arabic/intro.mspx
3. References on OpenType layout tags, Uniscribe, etc.: http://www.microsoft.com/typography/SpecificationsOverview.mspx
4. 《信息技术 藏文编码字符集键盘字母区数据区的布局》国家标准研制技术报告
5. THDL Extended Wylie Transliteration Scheme, THDL, University of Virginia
6. Microsoft Tibetan Keyboard Design Document, Tashi Tsering, China Tibetology Research Center
7. How to Use Microsoft Tibetan Keyboard, Tashi Tsering, China Tibetology Research Center
8. Windows Vista: http://www.microsoft.com/WindowsVista
9. Font Design Document for Microsoft Himalaya, Tashi Tsering, China Tibetology Research Center