1Q－1 **Sequencial Talks on the Multilingual Text Processing (1):**

Multilingual Processing of Historical Scripts with the Current Scripts

Yutaka Kataoka*, Tomoko Kataoka*, Kazutomo Uezono†,
Dawa Yidemucao†, Toshio Oya†, Hidejiro Daikokuya† and Hiroyoshi Ohara†

* Media Network Center, Waseda University † School of Science and Engineering, Waseda University

1. Introduction

In order to complete all the **World-wide Computings**, i.e, Digital Libraries, Voice Recognitions, Human Researches, Museum Database, etc., an Internationalized and Multilingual Computing Environment (IMCE), which can process all the scripts, including historical Scripts is essential. POSIX Locale model [1] cannot satisfy those above. By the analyses of all scripts, the Waseda IMCE, named ***The System 1***, was established as an ideal solution for the computings beyond ISO specifications and national standards. The System 1 broke the borders of Character codesets and differences of writing styles/orthographies, and brought processings of the real world (Fig. 1) by using of the real characters.

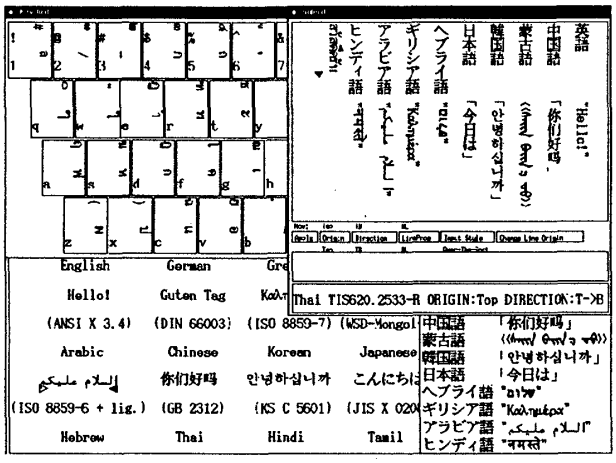

Figure 1. Running System 1

The system 1 was provided by *Global IOTMC System* that realizes an overall Internationalized and Multilingual Computing Environment including I/O, interprocess communication, text manipulation, programming languages and Internationalized X11 (Waseda X11) with interface compatibilities to National/International Standards [2]. By the discovery of the definition of *Character* [3], the kernel of the system named *Meta-Converter System* [3] not only absorbs dependencies of characters, codesets, languages and others but also realizes generalized text processings with specific information of such dependencies for advanced processing [4]. The system 1 is programmable to enhance codeset handling, Look & Feel and text processing rules by adding *Data Tables* which are compiled by *Data Table Compilers*.

2. Definitions of Internationalization and Multilingualism

All text processings have been considered as orthography and/or language dependent. But it was possible to separate text processing into two types; 1) Language-non-specific and 2) Language-specific [5]. And most of text processings can be Language-non-specific by adding a ***Character Specific Information*** into Character Representation as WC [5]. Thus, basic text manipulations, i.g., insertion, deletion, search, etc., can be Language-non-specific by such WC, which is a set of ***Character*** converted form *codesets* with character specific information. Therefore, Internationalization can be defined as processing all characters in one set of characters with Language-non-specific functions.

On the other hand, language-specific processings, i.g., hyphenation, spell-checking, requires language information in character itself. To operate such processings for Multilingual Texts, it is clear that each character should have language information [5]. Note that POSIX does not require such language-specific processings.

Thus, Localization is a subset of Internationalization and Multilingualism. But hyphenation and/or spell-checking can be operate by simple filtering to extract specific codeset(s).

3. Essentials of Characters

A ***Character*** was considered as a glyph or a grapheme. But a unit to be processed as a character is not a glyph or a grapheme (Fig. 2).

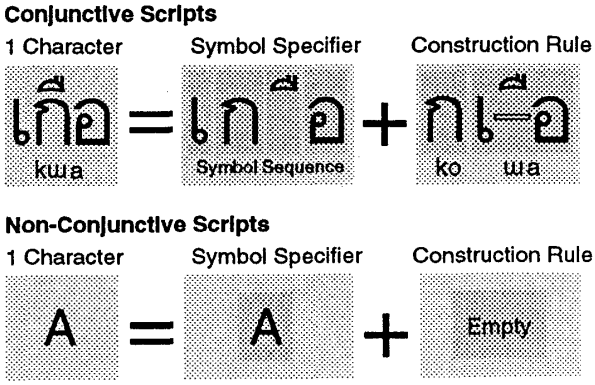

Figure 2. Internal Structure of Characters

The ***Construction Rules*** can be defined to generate a single set of characters by our researches [2, 4].

A shape of a character, i.e., glyph, is changed according to its position in a word and writing direction. Thus, character specifies a set of glyphs. By this, a character can be defined as a name of a set of glyphs (Fig. 3) [2].

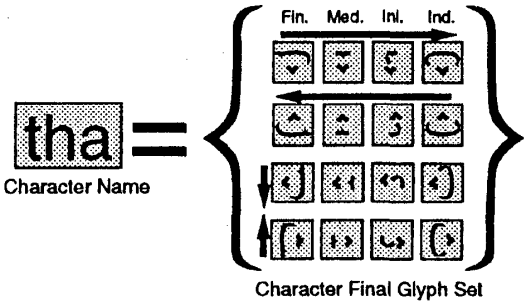

Figure 3. A Character represents a Glyph Set

Thus, function Drawing can be defined as selecting a suitable glyph in a set of glyph. To select one correct glyph by the function Drawing, writing conditions (at least writing direction and position in a word) should be determined.

4. Requirements to Be Realized

First of all, mixing all characters requires to draw vertical and horizontal writing scripts (Fig. 4).

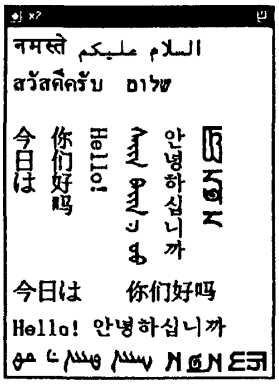

Figure 4. Mixing All Writing Directions

Ogham Script is drawn from bottom to top (Fig. 5). Thus, all four drawing directions should be supported.

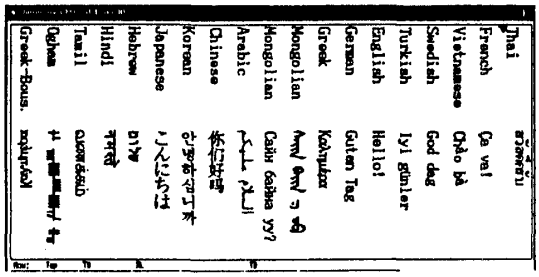

Figure 5. Ogham Drawn from Bottom to Top

Realizing ***Boustrophedon*** (Fig. 6) is essential for *Classic Greek* and *Hieroglyph.*

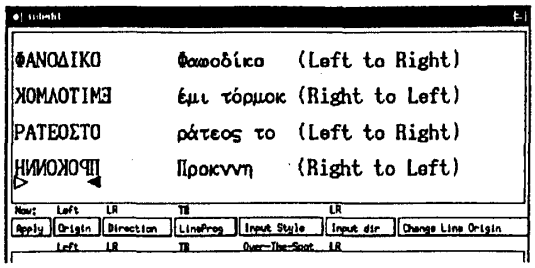

Figure 6. Boustrophedon in Greek

The examples above are only major requirements which should be realized at least. By such examples above, it is clear that WC should have fields to specify information for characteristics. Since the information can categorize characters, it is possible to normalize calling conventions of processing as function with attribute(s) in the information [4]. Also it is now clear that GUI for all scripts require combinations of rectangles for specific directions in a Window.

5. Summary

It was possible to extract essential information from historical scripts by the careful analysis. And the researches to all scripts including historical scripts could bring overall normalizations for calling conventions of processing functions and GUI.

Using The System 1, new trials to set differences among spoken languages and written languages can be started. Especially *Tibetan* writing is far from current spoken Tibetan. In this case, conversion from spoken language to written language is a major point for voice recognition. Spoken languages commonly contain older languages that carry extra information for parsing. Devanagari Scripts have special consonantal ligatures that show such information. And natural conversations often contain multiple languages.

Therefore, The System 1 permits us to process and to research natural languages as they are.

References

[1] ISO/IEC 9945-1: 1990, Information technology – Portable Operating System Interface (POSIX) Part 1: System Application Program Interface (API) [C Language].

[2] Uezono, K., et al., The Worldwide Multilingual Computing (2): Functions, Model, Design and Architecture of Multilingual I/O TM/C System, Proceedings of the 51th General Meeting of IPSJ, Vol. 3, September 1995, pp 247-248.

[3] Kataoka, Y. et al., The Worldwide Multilingual Computing (1): Essentials, Principles and Scope Covering All Characters in the World, Proceedings of the 51th General Meeting of IPSJ, Vol. 3, September 1995, pp 245-246.

[4] Kataoka, T., et al., The Worldwide Multilingual Computing (4): Essentials for the Multilingual Text Manipulation, Proceedings of the 51th General Meeting of IPSJ, Vol. 3, September 1995, pp 251-252.

[5] Kataoka, Y. et al., Internationalized Multilingual System - The Waseda I18N & ML System, Digital Libraries, Vol. 6, February 1996, pp 22-31, ULIS.