Building an Integrated Environment for Field Data Creation-Maintenance-Analysis: An XML & Unicode -based Workbench for Field Linguists

Shoju CHIBA

Reitaku University
College of Foreign Languages, Reitaku University
Hikarigaoka 2-1-1, Kashiwa, 277-8686, Chiba, JAPAN
schiba@reitaku-u.ac.jp

Abstract

Field linguists encounter various problems when they create their own electronic text data and try to utilize them for their research. This paper tackles two of the most serious difficulties, namely phonetic transcription and structured data description, and shows that introducing XML and Unicode may best promote the integration of fieldwork and data creation.

1. Introduction

This paper reports an on-going project to develop a fieldworkers' toolkit (*fwtk*, in short) for the textual study of endangered languages.

The *fwtk* project belongs to the ELPR Project, which is a four year-long academic project funded by the Japan Ministry of Education, Culture, Sports, Science and Technology and started up in 1999. ELPR means "Endangered Languages of the Pacific Rim" and the aim of it is to "engage in systematic and swift measures to document [the endangered] languages of the Pacific Rim" (from ELPR Web page). So far over a hundred linguists (including foreign researchers) have participated in this nation-wide project. The ELPR project includes seven research units, of which 4 (A01—A04) are regional studies and 3 (B01—B03) are multi-disciplinary. The *fwtk* project is a subproject in the Research Unit B03, which handle information processing issues (see Acknowledgements for further details).

The main aim of the *fwtk* project is to prototype a field linguists' toolkit which:

- enables you to handle various linguistic annotations, e.g. grammatical, semantic or pragmatic descriptions and phonetic transcriptions.
- focuses on portability, usability, and fills the special needs of field linguists:

The prototype *fwtk*, which is currently under development, is written in *Tcl/Tk* 8.3 and the target environment is Microsoft Windows2000/XP. The project URL is: http://www.fl.reitaku-u.ac.jp/~schiba/fwtk/ (now under construction), from which the update information of the tools will be available. A research workshop by B03 is being planned in the autumn and a manual (together with the software package) will be published as a publication of the ELPR Project by the spring, 2003.

Before proceeding to the details of the *fwtk* project, let us briefly review the state of the arts of the current computer-aided field research.

2. Motivations to Develop *fwtk*

There are two major problems that field linguists have been encountered when they create textual data, namely,

1. There are few straightforward ways to transcribe phonetic symbols. [1]
2. Once you decide to add some descriptive data to the texts, for example phonetic transcriptions, grammatical explanations, or notes on semantics, etc., the original texts may easily be intermingled with the additional data you have added. It is often true that the more annotations you add, the more difficult it becomes to retrieve original lines.

These problems have led linguists to a use of general purpose word processor: they can store various characters by choosing different fonts and indicate descriptive notes by different font styles, for example (Antworth & Valentine 1998: 171).

The data thus created would be sufficient for *printing*, but not so much for *processing*. The file formats are designed for a particular word processor, thus highly application-dependent, but most of the text processing tools, including widely known *grep* and *kwic* software, are applicable only to the plain text format. If descriptive levels are only distinguished visually, programs are very hard to recognize such information. [2]

The importance of avoiding word processor and storing data in plain text format has actually been well acknowledged. Nevertheless, there seems to be lacking any tools suitable for the analysis of field linguistic data: How IPA or other symbols can be represented? Or how can one systematically distinguish different levels of description and the raw data?

There are also special needs to enrich data processing environment for field linguists. In order to utilize IPA symbols in their text, they need a relevant support for manipulating these symbols. To handle various levels of de-

[1] There have been proposed alternative ways to represent IPA in plain text (SAMPA project since 1987; Kirshenbaum 2001, for example), but they require further training to read and edit the data and additional skills in processing texts.

[2] It is worth mentioning here that there is a growing interest in publishing resources on the Web or on CD-ROM; there is even a trend to publish multimedia resources of field data on CD-ROM (Nathan 1999). Making machine-readable data for endangered languages and publishing them electronically are thus becoming more and more essential among researchers' tasks.

scriptions, again, they need a special structure description format and also good facilities for structure-sensitive search and text view.

In the corpus-linguistic tradition, there have been developed several linguist-oriented tools, among which LEXA (Hickey 1992, 1994) and TACT (Lancashire 1996) may be the most widely used. These tools are in themselves very powerful, and field linguists, too, might well be good candidates for the potential users of them. However, these tools run only on the DOS environment and the interface is highly specialized, there seems to be unfortunately little room for introducing them to the workspace of field linguists.

3. Designing Features of *fwtk*

The prototype *fwtk* implements two recent technologies to solve the problems mentioned above: Unicode and XML. It also includes the tools and interfaces designed for the field linguistic study.

3.1. Maximizing integration

Field linguists, especially working with endangered languages, make a great effort at managing the data they collect: actually many of them document, analyze, and publish the data by their own efforts. This means that once the data is digitalized, they will have to master different tools according to the different phases of their job. This often causes difficulties, as the programs differ in their user interface and the functions.

Here is the reason why an integrated workbench designed for field linguists is needed: there should be a package which covers most of their basic tasks and shows the maximum integrity with regard to the user interface and functionality.

Specifically, our *fwtk* includes the following tools:

- Basic tools for different phases of the field study
  - ✓ A Tool for Data Creation (Edit), which
    - ♦ shows the full data with XML tags, and
    - ♦ assists in editing text with XML tags
  - ✓ A Tool for Data Management (Arrange), which
    - ♦ shows the text without XML tags, and
    - ♦ lists the linguistic descriptions on the textual level selected
  - ✓ A Tool for Data Analysis (Search), which
    - ♦ shows the text without XML tags, and
    - ♦ hides linguistic descriptions (which can be called through a pop-up window)
- Tools repeatedly used with the toolkit
  - ✓ IPA Soft Keyboard
  - ✓ Basic Search function, which implements Unicode characters and their character reference counterparts

The basic tools should be maximally integrated, so that one can call any basic tools when you want and proceed swiftly from one tool to another. This is essential, for fieldworkers often do the three tasks (editing, analyzing, maintaining) either in succession or even simultaneously.

3.2. Using Unicode for Text Encoding

Unicode (Graham 1999; Unicode Consortium 2000) is a new character encoding standard published by the Unicode Consortium and is "designed to include all of the major script of the world in a simple and consistent manner" (Graham 1999:75). Unicode is thus fully multilingual and includes full IPA symbols.

fwtk takes full advantage of Unicode and store the data in ***utf-8***, a transformed format of Unicode suitable for data exchange via network.

Because the structure of Unicode is fundamentally different from the existing encoding systems, the compatibility issue may be a potential problem when we use Unicode (see §3.6. for the details). To avoid any loss of the data in a Unicode-unaware program, *fwtk* has an export tool which converts Unicode characters to their Numeric Character Reference equivalents (notation: &#x*nnnn*; where *n* stands for hexadecimal number of the Unicode code point). Furthermore, in the searching and the text view functions, *fwtk* treats Unicode characters and their numeric reference counterparts as equivalent expressions.

The IPA inputting system of *fwtk* has 3 ways of character display, namely:

- Tables implementing IPA charts (International Phonetic Association 1999)
- Character list sorted by IPA number or Unicode order

This IPA keyboard dialogue can be run anytime from any main tools of *fwtk*. The Figure 1 shows how the software keyboard looks like:

Figure 1. IPA Software Keyboard

3.3. Using XML to Describe Text Structure

XML (eXtensible Markup Language) is a modern markup method to express data structure in plain text format, which is intended to be simple and explicit to process. It is extensible in the sense that one can define his/her markup tags according to his/her needs. XML is proposed by World Wide Web Consortium and the current version is 1.0 (Second Edition, issued in October, 2000).

fwtk implements XML so that the data is processed and stored in XML format. It also provides import/export functions, which are summarized below (see Figure 2):

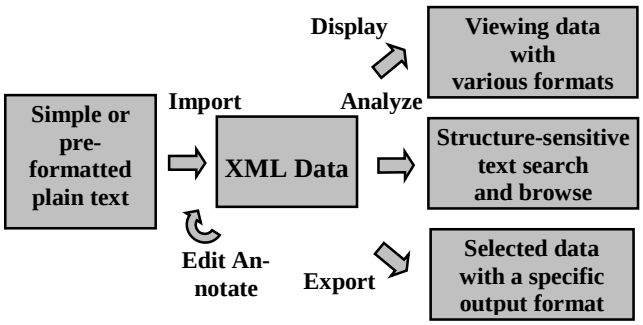

Figure 2. How *fwtk* utilizes XML

3.4. Structural Features of Field Data

Once you decide to implement XML, then you can freely define, how complex the data will be: XML is *eXtensible*. Structure of field data varies greatly according to the contents, but as far as the textual data is concerned, it falls into two main categories:

- Word list, which can be expressed in simple 2-dimensional table
- Phrase list or sequential text (narrative text, for example), which is more complex and can't be expressed by a simple table-like format

Because a spreadsheet-type program (like Excel) is well applicable to the former type of data, it is the latter type of textual data that *fwtk* project is particularly focusing on.

Now the question is: to what extent a normal field-linguistic data should be complex. We need to distinguish two levels of complexity:

- Structural level: how each text element is arranged and grouped. Usually, one wishes to distinguish, for example, *Paragraph*, *Sentence*, and *Word* level.
- Descriptive level: how each text element (*Word*, for example) can be linguistically described. For linguistic purposes, we need to annotate grammatical description (e.g. *Part of Speech*, *Stem*/*Root*, basic form (*Lemma*), *Syntactic Role*), phonetic transcription, and other descriptive memos, etc.

With the distinctions mentioned above in mind, in the prototype program we restrict the target data structure as simple as possible and represent it in XML format as follows:

- Elements (which indicate the structural level information)

  <body>text body</body> … shows the range of main text
  section ... shows the main divisions that the main text embodies
  paragraph
  <s>sentence</s>
  <w>word</w>

- Attributes (which indicate the descriptive level information and annotate elements)

  <w phon="phonetic transcription" lemm="basic form" gram="grammatical description" memo="descriptive note">word</w>

The following Figure 3 shows a sample of the target data structure of *fwtk*:

```
<body>
  <div>
    <p>
      <s memo="speaker A">
        <w gram="POS" lemm="basic form"
           phon=" IPA " >WORD</w>
        …
      </s>
      <s>… </s>
    </p>
  </div>
  <div>
  …
  </div>
</body>
```

Figure 3. Target Data Structure of *fwtk*

3.5. Structure-Sensitive Search

As is briefly mentioned in §3.2, the search function of *fwtk* supports Unicode and its character reference notations. However, this isn't actually sufficient for the linguistic analysis. *fwtk* enriches the search function with the following features:

- Implementation of regular expression
- Structure-sensitive search method designed for each display mode
  - ✓ to keep the output format identical with the original text displayed
  - ✓ to enable to specify the search field by Element/Attribute

The latter feature is particularly important to successfully skim off a pattern on a specific descriptive level and arrange it for display.

Further, on the Search window two types of data display method are available.

- Enhanced *grep*, which
  - ✓ specifies a string and the descriptive level where it occurs, and
  - ✓ searches the string and shows the sentence(s) which include it
- Enhanced *kwic*, which
  - ✓ searches a string (with a particular attribute value), and
  - ✓ shows it with a range of context

3.6. Future Development

Because *fwtk* is still under development, there still remain functions unincorporated into it. There are also several technical problems to be solved during the development of the application. Here is a list of the general problems we are confronting:

1. Full implementation of Unicode is technically very requiring.
   - ✓ **Implementation levels:** Degree of implementation varies between OS versions or programming languages. For example, A *Tcl/Tk* program featuring Unicode slows down on Windows9x platform.
   - ✓ **"Dynamic composition":** Combined characters with multiple diacritical marks are open-ended. This causes difficulties for displaying and printing complex characters correctly.
   - ✓ **One combined character, many representations:**
     - ♦ There can be multiple ways to express a particular character: One can represent a character with a single character or a combined character string. A program should aware the correspondence of the different expressions, and it requires the detailed inventory of such relationships.
     - ♦ There are wide varieties of symbols that have separate character codes assigned and look nevertheless similar. This easily leads to the inconsistency of the characters used in the data.
2. The prototype *fwtk* treats XML on rather unsystematic basis and doesn't support any customized tags. There is a room for improving efficiencies of XML data parsing. Note also that the user interface for editing XML is experimental and to be re-designed in the future version.
3. Language-specific customizations remain unimplemented. For example, a sub-program that sorts the search results by a language-specific order would be highly desirable.

4. Conclusion

This paper introduced an Unicode & XML -compliant toolkit designed for field linguists and examined how these two technologies, when tightly united, can facilitate the creation, maintenance and the analysis of field linguistic data.

Bringing together the various functions needed for field linguists, a small but well field-oriented software toolkit like *fwtk* will make individual researches more efficient, simplify the process of the publication of the data on various media (on the Web or CD-ROM), and facilitate the vigorous exchange of the data between researchers.

5. References

Antworth, E. L. and J. R. Valentine, 1998. Software for doing field linguistics. Lawler, John and Helen Aristar Dry (eds.) *Using Computers in Linguistics: A Practical Guide*. London: Routledge. pp. 170—196.

ELPR (Endangered Languages of the Pacific Rim), Endangered Languages Project: Outline of Research. http://www.elpr.bun.kyoto-u.ac.jp/outline_e.html

Graham, T., 1999. Unicode: what is it and how do I use it? *Markup Languages: Theory and Practice* 1: 75-102.

Hickey, R., 1992. *Lexa: Corpus Processing Software.* Vol. 1—3. Bergen: Norwegian Computing Centre for Humanities.

Hickey, R., 1994. *Lexa Version 6.0: Update Documentation.* Bergen: Norwegian Computing Centre for Humanities.

International Phonetic Association, 1999. *Handbook of the International Phonetic Association: A Guide to the Use of the International Phonetic Alphabet*. Cambridge: Cambridge University Press.

Kirshenbaum, E., 2001. Representing IPA phonetics in ASCII. http://www.kirshenbaum.net/IPA/ascii-ipa.pdf

Lancashire, I., 1996. *Using TACT with Electronic Texts*. New York: Modern Language Association of America.

Nathan, D., 1999. Tools for communities, tools for linguists: new technologies for endangered languages. Paper read at the ICHEL Colloquium on Endangered Languages, Oct. 27, 1999, Tokyo University, Japan.

SAMPA (Speech Assessment Methods Phonetic Alphabet), Computer Readable Phonetic Alphabet. http://www.phon.ucl.ac.uk/home/sampa/home.htm

Unicode Consortium, 2000. *The Unicode Standard Version 3.0.* Reading, MA: Addison Wesley Longman.

6. Acknowledgements

This project is a sub-project of the Research Unit B03 of the ELPR project ("Digitalization of linguistic data and information retrieval for the study of endangered languages"), funded by Ministry of Education, Culture, Sports, Science and Technology (Grant-in-Aid for Scientific Research on Priority Areas [KAKENHI] No. 12039213).