Conversion of Microsoft Word and OpenOffice formats into xml-like documents

Martin Šmotlák[†] and Rudolf Hladík[†]

November 26, 2004

[†]*Department of Slovak National Corpus, L'udovít Štúr Institute of Linguistics, Slovak Academy of Sciences, Bratislava, Slovakia*

Abstract

In this technical report the conversion of most frequently used document formats into the specified xml-like format is discussed. The developed conversion software package is presented and its use is described. It was created for purposes of building the Slovak National Corpus of written language. This package is able to convert all the file formats recognized by Microsoft Word and OpenOffice writer, which are nowadays the wide-spread text editors.

1 Introduction

During the last years corpus linguistics has received a big attention and draws many linguists. Building of corpuses is also an interesting task not only for linguists. From a technical point of view the process of building of corpus is constrained mainly to a document processing and converting it to uniform standard text form. Useful and commonly used format is the `xml` one. It can express all properties of documents (like titles, paragraphs, font styles,...) on the text level. These characteristics are represented by the xml tags. In the implementation of the Slovak National Corpus (SNC) [1] xml-like structure is postulated as an unified form of assembled corpus texts. So one has to convert incoming documents into that defined xml style.

The documents from text providers are collected in many various formats. In better case a text donor is able to provide an electronic version of desired document. This concerns mainly the documents from the last decade. In the case the document is not available in electronic form, it should be scanned – converted into electronic data, usually preserved as a **tiff** file. Then the data are recognized by FineReader and stored e.g. in **rtf** or **sxw** format. The procedure of making electronical data from written text will be described in more details in chapter 2.

From technical aspect of view the archive of SNC contains many various types of documents. The most frequent are *raw texts*, *t602* and *html* files and also *doc*, *rtf*, *sxw* documents. Conversion of plain texts, t602 and html files is not so difficult, because existing xml tools correctly convert also diacritics. In this contribution we will describe how to convert all documents recognized by Microsoft Word (MS Word) so as to keep correct identification of diacritic characters.

2 Conversion of written text into electronic data

In the department of SNC the book scanner *Minolta PS 7000* is used for scanning of texts. The recognition of scanned text is made by the OCR software *ABBY FineReader*. Recognised text is usualy corrected in the Microsoft Word or in the Open Office text editor.

2.1 Scanning procedure

Up to now only two page 5 is made

3 Conversion of documents recognized by MS Word

The problem of todays Microsoft Word converters into texts (e.g. *wv tools*) is their possible failure in diacritic characters conversion. It depends on the version of MS Word in which the considered document was created or exported. The files produced in old versions of Microsoft Word[1] are not correctly converted because the existing converters automatically assume one type of encoding[2] if this information is missing in a file (this is just the case of the old MS Word documents). The solution of this problem is to load the document into one of the newer versions of MS Word, which is able to recognize the correct encoding, and save the document there. This procedure ensures the correct character encoding in conversion process. Below the proposed conversion procedure will be described in more details.

[1] up to MS Word 97 including

[2] namely *cp1252* (Western European) encoding

3.1 Software requirements

- operating system: *Windows*
- necessary applications: *Microsoft Word* (MS Word) version later than 97, *OpenOffice*, *ActivePython* with module *ElementTree*

3.2 Conversion steps

The conversion programs are written in Python. They were developed for conversion of all types of documents, which are reasonably readable by MS Word (e.g. files with extension **doc**, **rtf**, **wri**, **txt**...). The procedure of conversion is as follows:

1. The file which is a subject of interest, is opened in the MS Word environment and exported in `doc` format (MS Word default format). As explained above, this is done due to character encoding identification and its correct indication in the head of converted file (in the case it is missing there).
2. Second step comprises conversion of `doc` file type into `sxw` one. As the OpenOffice writer knows **doc** format, this conversion is done by reading a `doc` file in OpenOffice environment followed by storing it in the **sxw** format. **sxw** is a standard file type (intrinsic format) of text editor in the package OpenOffice. The file with extension `sxw` is actually a packed set of `xml` files compressed by the **zip** archiver. This `xml` files contain the whole information about document (head, style, fonts, ...). The body of document is stored in a separate file named `content.xml`.
3. The last step of conversion consists of an extraction of `xml` text (*content.xml*) from the `sxw` file. In this process original xml tags are also modified due to the xml convention defined for SNC [1] and unnecessary tags in text are omitted.

Additional step of conversion can be assignment of all types of titles with corresponding begin and end tags `<h1>` and `</h1>` respectively (see Ref. [1]). This action is not a part of described conversion process because it depends on concrete document properties and should be done separately for each document with a conveniently modified text parser.

It is worth to mention that the described procedure allows to convert directly **sxw** documents. One can just leave out the first two steps of conversion process in a main program.

3.3 Code description

The package for conversion contains three partial programs `any2doc.py`, `ooextract.py` and `oo2bank.py`, which execute the conversion steps from previous section 3.2. Namely:
– `any2doc.py` opens the converted file in MS Word and save it as a doc file (step 1),
– `ooextract.py` load the produced doc file into OpenOffice writer and store it in internal sxw format (step 2),
– `oo2bank.py` extracts the body of stored sxw document into xml file (step 3).
Introduced subprograms are called in main Python program `parse.py`. The main code scans the SNC archive directories and looks for files with a given extension (e.g. doc). If it finds a file of desired type, the conversion is executed and the converted file is stored in a defined working directory with a copy of full path to the converted document in archive (for details see Appendix). Scanning of archive proceed further then. The main program is terminated when the all archive (or its given part) is overlooked.

The program `ooextract.py` which makes the conversion $\mathrm{doc} \rightarrow \mathrm{sxw}$, requires running of OpenOffice writer with some specific parameters. This ensures the script `prepare.bat`, which should be executed before the start of conversion (parse.py). The file `prepare.bat` contains two command lines:

```
cd c:\Program Files\OpenOffice.org1.1.0\program\
soffice "-accept=socket,host=localhost,port=2002;urp;"
```

which serve for correct launching of the OpenOffice. It is recommended to put all the program files into one working directory and create separate directory for storing the converted files (as it is done in an example case in Appendix).

4 Conclusion

In this technical report the conversion of written text into electronic data was described. The conversion procedure consists of the three steps. They were characterized in more details and the practical experiences from their realization were also mentioned. Further we have introduced the principle for correct converting of Microsoft Word documents into xml format. Described procedure provides conversion of all document types recognized by MS Word independently on the version of MS Word in which the document was created. The realization of proposed conversion principle was outlined and the procedure was used for converting of documents from the archive of the Slovak National Corpus. This work was stimulated by needs of building and enlarging of the general corpus of Slovak written language. The developed conversion package can be used and considered as a new version of xml tool for exporting the MS Word files into xml ones.

A Appendix

One possible realization of a main code, which walks through the document archive and performs the conversion of given type of documents.

```
import os
import sys
from os.path import join, split, splitext

def batch( meno ):
   import os
   import sys
   from os.path import splitext
   import any2doc

   # make of correct *.doc file
   os.chdir('c:/xmlconv/work')
   any2doc.convert('c:/xmlconv/work',
                    meno, '__text__.doc')
   # Conversion from *.doc to *.sxw
   # makesxw executes the code ooextract.py
   # in OpenOffice own python
   os.system('c:/xmlconv/makesxw')
   # Conversion from *.sxw to *.xml
   os.system('c:/Python23/python.exe
      c:/xmlconv/oo2bank.py __text__.sxw __text__.xml')

os.chdir('c:/xmlconv/work')
cast_archivu = r'\data\snk\archive\2004'
for data in os.walk( cast_archivu ):
   root = data[0]
   current_dir = root[len(cast_archivu):]
   if len(current_dir) <> 0:
       dir = current_dir[1:]
       os.mkdir(dir)
   for file in data[2]:
     ext = os.path.splitext( file )
     if ext[1].lower() in ( '.rtf', '.wri' ):
```

```
name = os.path.join(root,file)
print 'converting:' file
batch( name )
print 'OK:', file, name
```

References

[1] Garabík, R. Štruktúra dát v Slovenskom národnom korpuse a ich vonkajšia anotácia: Proceeding from conference *Slovenčina na začiatku 21. storočia*: Prešov, 7.3.2003.